1. GİRİŞ

Günümüzde hızla gelişen teknoloji insan hayatını kendi etkileri oranında değiştirmektedir. Özellikle bilişim teknolojilerindeki gelişmeler; gerek hızı gerekse etkinliği itibariyle, çağımız insanının iş, iletişim, bilgi edinme gibi birçok alışkanlığını temelden değiştirdi. Gelişmiş bilgisayar sistemleri ve bu sistemler arasındaki İnternet, küresel bazda, hemen hemen her çeşit bağlantıya imkan sunmaktadır. Evinizin bir odasındaki bilgisayarınızda siparişlerinizi verebilir, bankacılık işlemlerinizi yapabilir, her türlü ses, veri ve görüntü transferini gerçekleştirebilirsiniz.

Bilgiye ulaşmanın ve bilgi dolaşımının bu kadar kolay olduğu bir ortamda ,şüphesiz ki en büyük sorun, "dolaşımda olan bilginin güvenliği" sorunudur. Gizlilik mahiyeti yüksek bir bilginin iletimi, önemli miktarlardaki para transferlerinin gerçekleştirimi gibi üçüncü kişilerin bilgisine kapalı kalması zorunlu bütün durumlar için elektronik ortamların (İnternet/intranet) güvenirliği son derece önemlidir. Günümüz şartlarında, kişi-bilgi etkileşiminin niteliği değişmiş olup; bilginin elde edilmesinden çok, bilgiyi kullanmanın önemi artmıştır. Kötü niyetli bir İnternet kullanıcısının, size ait banka hesapları bilgilerine ulaşarak, hesabınızı kendi kontrolüne alması buna verilebilecek çarpıcı bir örnektir.

İnternet ortamının, dolayısıyla elektronik ortamların kolaylıklarından yeterince faydalanabilmenin şartlarından biri de; elektronik ortam olarak İnterneti, günlük kullanım alanlarına gerektiği kadar yayabilmektir. İşte tam bu noktada, İnternet ortamında yapılan işlemlerin hukuksal olarak geçerlilik niteliğine sahip olması problemi önem kazanıyor. Çünkü o zaman gerçekleştirdiğiniz işlemin, gerçekleşme kolaylığı ile hukuksal olarak geçerli olma özelliklerinin birlikteliği sağlanmış olur.

İnternet üzerinde dolaşan bir bilginin hukuksal açıdan geçerli olması ve güvenliğinin sağlanması için; gizlilik, verinin şifrelenmesiyle, bütünlük, özetleme algoritmalarıyla, kimlik doğrulama ve inkar edilmezlik özellikleri de sayısal imza ve sayısal imza ile ilgili işlem kayıtlarıyla sağlanır. Bu saydığımız özellikleri itibariyle

çalışma konumuz olan elektronik imza, bilgiyi kimliklendirme açısından, son derece önemli bir yere sahiptir.

Özellikle e-devlet uygulamalarının hayatımıza kapsamlı ve etkili bir şekilde girmesi, günlük yaşamda resmiyet ihtiva eden iş ve işlemlerin hızlı, güvenli ve hukuksal olarak geçerli mahiyette oluşturulması, e-imza ile mümkün olmaktadır.

Bilgi toplumu olma hedefinde, bilgi güvenliğinin sağlanması, etkin e-imza uygulamaları ile e-devlet, e- ticaret alanında yapılan iş ve işlemlerin hızlı, kaliteli ve güvenli bir şekilde zamandan ve mekandan bağımsız olarak gerçekleştirilebiliyor olması, bu çalışmanın esas amacını oluşturmaktadır. Genel olarak e-imza kapsamı ortaya konulduktan sonra, e-imza oluşturmada kullanılan kriptografik algoritmalar incelenip, kendi aralarında, olumlu ve olumsuz açısından karşılaştırılmıştır.

BİRİNCİ BÖLÜM
GENEL OLARAK İMZA ve GÜNÜMÜZDE KULLANILAN ISLAK İMZANIN İŞLEVİ

1. Genel Olarak İmza

Günlük yaşamda herhangi bir belgeye resmi nitelik kazandırmak için kullandığımız imzaya ıslak imza denir. Islak imza; bir kimsenin bir yazının altına, bu yazıyı yazdığını yada onayladığını belirmek için, her zaman aynı biçimde ve kendi eliyle yazdığı, kendi adı yada adının imidir.[1]

Ayrıca Türk Hukukunda ıslak imza, ilk defa 22 Nisan 1926 tarih ve 818 sayılı Borçlar Kanunun 14.Maddesinde şöyle tanımlanmıştır; ”İmza,üzerine borç alan kimsenin el yazısı olmak lazımdır. Bir alet vasıtasıyla vazolunan imza, ancak örf ve adetçe kabul olunan hallerde ve hususiyle çok miktarda tedavüle çıkarılan kıymetli evrakın imzası lazım geldiği taktirde kafi olunur. ”

Tanımından da anlaşılacağı üzere imza; imzalanan belgenin, imzalayan tarafından, inkar edilmeme, doğrulama ve bütünlüğünü sağlama unsurlarını güvence altına alır. Dolayısıyla kişi imzaladığı belgeden doğabilecek bütün sonuçlardan hukuken sorumludur.

1.1 E-İmza

E-imza; gelişmiş teknolojiler kullanılarak, elektronik ortamda gönderilen veya alınan bilgilerin bunları gönderen kişi veya kuruma ait olduğunun doğrulanmasını, iletilen veya alınan verilerin bilinmeyen kişiler (başkaları) tarafından gönderilmediğinin belirlenmesini , verileri gönderenlerin gönderdiğini ve alanların aldığını inkar edememesini, gönderilen veya alınan bilgilerin içeriğinin değiştirilmemesini, başkaları tarafından elde edilse bile, içeriğin başkaları

tarafından anlaşılmamasını sağlamayı garanti eden, elektronik ortamda bit katarlarından oluşturulmuş güvenli haberleşme ortamına verilen addır.[2]

5070 sayılı Elektronik İmza Kanunu'nda yer alan şekliyle elektronik imza; başka bir elektronik veriye eklenen veya elektronik veriyle mantıksal bağlantısı bulunan ve kimlik doğrulama amacıyla kullanılan elektronik veriyi tanımlar. Elektronik imza; bir bilginin üçüncü tarafların erişimine kapalı bir ortamda, bütünlüğü bozulmadan (bilgiyi ileten tarafın oluşturduğu orijinal haliyle) ve tarafların kimlikleri doğrulanarak iletildiğini elektronik veya benzeri araçlarla garanti eden harf, karakter veya sembollerden oluşur. Elektronik imza kavramı çok genel bir tanım olup kişilerin elle atmış olduğu imzaların tarayıcıdan geçirilmiş hali olan sayısallaştırılmış imzaları, kişilerin göz retinası, parmak izi yada ses gibi biyolojik özelliklerinin kaydedilerek kullanıldığı biyometrik önlemleri içeren elektronik imzaları veya bilginin bütünlüğünü ve tarafların kimliklerinin doğruluğunu sağlayan sayısal imzaları içermektedir. Sayısal imza, imzalanan metine göre farklılık gösterir ve içeriğin matematiksel fonksiyonlardan geçirilerek eşsiz olduğu düşünülen bir değer bulunması sureti ile elde edilir. Yani kişilerin, elle atılan imzada olduğu şekilde tek imzası yoktur; bunun yerine imzalamada kullanılan anahtarları vardır.

5070 sayılı Elektronik İmza Kanunu'nda ve bu metinde geçen "elektronik imza" kavramı sayısal imzayı işaret etmektedir. [3]

Amerikan "Electronic Signatures in Global and National Commerce Act (E-Sign)", e-imzayı, elektronik bir ses, sembol veya veriye eklenen veya elektronik veriyle mantıksal bağlantısı bulunan verileri değiştirmek veya işlemek için kişinin verileri imzalama (işaretleme) girişimidir.

Yukarıda farklı otoritelerce yapılan tanımların ortak yaklaşımından bir e-imzada bulunması gereken önemli özellikler;

- Güvenirlik; kullanılan e-imzanın gizliliğinin uygun şifreleme yöntemleri ile sağlanmış olması,

- Taklit edilmezlik; e-imzanın tekil olması,
- Yeniden kullanılmazlık; e-imzanın tekrarlı kullanımlardan doğabilecek saldırılara kapalı olması,

- İnkar edilemezlik; e-imza ile işaretlenmiş veri için, gönderenin veriyi gönderdiğini, alıcının veriyi aldığını onaması,

- Bütünlük; gönderilen verinin içeriğinin yolda değiştirilmemesi,

- Kolay kullanabilirlik; genel olarak Dünyada ve özel olarak ülkemizde, bilgisayar okur-yazarlığının düşük düzeyde olmasından dolayı; yaygın e-imza kullanımı için, kolay kullanım yollarının tercih edilmesi, olarak sıralanabilir.[2]

Bilgi teknolojilerinin, özellikle bilişim alanındaki farklı gelişmeleri, e-imza tanımlarının kapsamını yeniden belirleyebilir. Özellikle biyometrik imzaların yaygınlaşması; iris tanıma, parmak izi ve genetik imzanın, matematiksel algoritmalardan elde edilmiş kriptografik yöntemlerden öte; sadece kişinin, kişisel /özgün (spesifik) niteliklerine bağlı olarak elde edilmesi, bu yaklaşım için iyi bir örnektir. Nitekim olası gelişmelere karşın, Türkiye'de kabul edilen e-imza kanunları çerçevesinde belirlenen e-imza standartları 31/12/2008 tarihine kadar geçerli kılınmıştır.

1.2 E-İmza Gereksinimi ve Kullanım Alanları

Hızla gelişerek yaygın hale gelen bilgi teknolojilerini kullanarak; bürokrasiyi azaltmak, kağıt belge formatından kurtulmak ve bu yönlü tasarruflarla çevreci bir yaklaşımı geliştirmek, yerel ve global düzeydeki iş akışlarına hız kazandırmak, zaman tasarrufunu sağlayarak verimli iş performanslarını elde etmek gibi bir çok açıdan e-imza kullanımı önemlidir.

5070 sayılı Elektronik İmza Kanunu'nda yer verilen hüküm itibariyle e-imza; evlilik, kefalet ve tapu işlemi dışında kalan diğer bütün kamusal ve ticari alanlarda kullanılabilecektir.

1.2.1 E-imzanın Kamusal Alandaki Uygulamaları

E-imza uygulamalarının toplumsal yaşayışımıza getireceği en önemli yenilik, e-devlet dönüşümünün, vatandaş ve devlet etkileşmelerinin kurumsal işleyişi noktasında, iş ve işlemlerin elektronik ortama taşınıyor olmasıdır.

- Her türlü başvurular (ÖSS, KPSS, LES, pasaport vb)
- Kurumlar arası iletişim (Emniyet Müdürlükleri, Milli Eğitim Müdürlükleri, Üniversiteler, Nüfus ve Vatandaşlık İşleri Müdürlükleri vb.)
- Sosyal güvenlik uygulamaları
- Sağlık uygulamaları (Sağlık personeli, eczaneler, hasta takip vb. uygulamalar.)
- Vergi ödemeleri
- Elektronik oy verme işlemleri gibi kamusal alandaki bir çok uygulamada kullanılmaktadır.

1.2.2 E-imzanın Ticari Alandaki uygulamaları

E-Ticaret kavramı, günümüzün ekonomik faaliyetlerinin vazgeçilmez bir uygulaması haline geldi. Nitekim, ülkemizde e-imzanın, e-ticarette kullanılması ilk olarak Dış Ticaret Müsteşarlığınca gerçekleştirilmiştir. Her yönüyle globalleşen Dünyamızda, ticaret de globalleşmiştir. Dünyanın her köşesiyle uluslar arası konjöktörde geçerli olan hukuk güvencisiyle ticari faaliyet sürdürmek e-imza sayesinde olanaklı hale gelmiştir.

- İnternet bankacılığı
- Sigortacılık işlemleri
- Kağıtsız ofisler
- e-Sözleşmeler
- e-Sipariş gibi bir çok ticari alanda e-imza kullanılmaktadır.

1.3 E-İmzanın Hukuksal Geçerliliği

E-imza ile İnternet üzerinden iş ve işlem gerçekleştirmenin avantajlarından faydalanabilmek için, e-imzanın sahip olması gereken önemli özelliklerinden başında, hukuksal niteliği itibariyle, ıslak imza ile eş değerde olmasıdır. Böylece e-

imza ile gerçekleştirdiğimiz bütün işlemler, hukuksal açıdan yasal geçerlilik kazanır. Günümüzde, Dünya'nın bir çok ülkesinde, e-imzanın yasal geçerliliğini öngören kanunlar oluşturulmuştur. Ülkemizde de aynı amaçla 5070 sayılı "E-imza Kanunu" 15 Ocak 2004 tarihinde kabul edilmiş olup, 23 Ocak 2004 tarihinde de 25355 sayılı Resmi Gazete yayımlanarak, yasallaştı. 23 Ocak 2005 tarihi ise, e-imzanın uygulaması için başlangıç tarihi seçildi. Bu çalışmanın sonunda "E-imza Kanunu" Ek-1' de ayrıca, e-imzayı oluşturma ile ilgili genel standartlar Ek-2' de belirtilmiştir.

1.4 E-İmza Konusunda Yapılan Çalışmalar ve Bu Çalışmalardan Elde Edilen Sonuçlar

E-imza ile ilgili çalışmaları detaylandırabilmek için, e-imzanın oluşumundan, kullanımına kadarki bileşenlerinden söz etmek gerekir. Bilindiği gibi e-imza bilgi güvenliği ve hukuksal geçerlilik kapsamında;

- Gizlilik
- Bütünlük
- İnkar edilmezlik
- Kimlik doğrulama elemanlarından oluşuyor. Dolayısıyla e-imza ile ilgili

çalışmalar, söz konusu bu dört temel elemanın gelişimi aşamasında yapılan çalışmalardır.

1.4.1 E-imza Oluşturmada Gizliliği Sağlayan Şifreleme Bilimi ile İlgili Yapılan Çalışmalar

Genel olarak bilgi güvenliği şifreleme algoritmaları ile sağlanır. Şifreleme; bir verinin, matematiksel fonksiyonlar yardımıyla istenmeyen kişilerce anlaşılmayan bir forma dönüştürülmesidir.

- İlk şifreleme yaklaşımlarının M.Ö 1900 lerde Mısırlılar tarafından kullanıldığı tespit edilmiştir.
- M.Ö 100-44 Yılları arasında Sezar tarafından kullanılan şifreleme, kullanılan ilk şifreleme olarak kabul edilir.
- 1623'te Francis Bacon, 5-bit ikili kodlamayla karakter tipi değişikliğe dayanan yaklaşımı geliştirdi.

- 1790 da Thomas Jefferson tarafından "strip cipher" makinesi icat edildi.
- 1917 de Joseph Mauborgne ve Gilbert Vernam "one time pad" algoritmasını geliştirdi.
- Enigma 1920 yılında ticari olarak kullanıldı. Fakat asıl önemli özelliği II.Dünya Savaşının sonucunu belirlediğine inanılmasıdır. Rotor makineleri ailesi ile ilişkili bir elektro-mekanik aygıt olarak yapılmıştı.
- II.Dünya Savaşından sonra Amerikada Ulusal Güvenlik Merkezi kuruldu. 1960 ta IBM'in çalışmaları başladı. Daha sonra 1970 te ABD Federal Bilgi İşleme Standardı benimsendi ve DES (Data Encryption Standart) oluşturuldu, e-imza ve e-ticarette kullanıldı.
- 1976 yılında Diffie ve Hellman tarafından ilk açık anahtarlı şifreleme yaklaşımı teorik olarak ortaya atıldı. Tam iki yıl sonra Rivest, Shamir ve Adleman adlı bilim adamaları tarafından RSA adıyla ilk açık anahtarlı şifreleme yaklaşımı gerçekleştirildi.
- 1990 yılında Lai ve Massey tarafından IDEA, akabinde 1991 de Zimmerman tarafından PGP (Pretty Good Privacy) geliştirildi.
- Sayısal imza konusunda ilk uluslar arası standart olarak ISO/IEC 9796 uygulamaya konulmuş ve algoritma olarak ta RSA kullanılmıştır.
- 1994 te ABD Hükümeti ElGamal algoritmasına göre oluşturulmuş açık anahtar yapısıyla e-imza standardını kabul etmiştir.
- En son olarak, günümüze kadar geliştirilmiş bütün şifreleme yaklaşımlarına alternatif olarak, Rejindal ve AES (Advenced Encryption Standart) algoritmaları geliştirilmiştir.[2]
- 1984 te Bennet ve Brassard BB84 Protokolü ile ilk "Kuantum Kriptografik Algoritmasını" gerçekleştirdiler.

Kriptografinin gelişim sürecinde 1976 yılında Diffie ve Hellman'a kadarki bütün yaklaşımlar simetrik şifreleme algoritmları olarak bilinir. Simetrik şifreleme algoritmaları tek gizli anahtarın kullanılması ilkesine dayanır. Bu yaklaşımda anahtar dağıtımı sorun olup, e-imza uygulamalarına uyumlu değildir.

1976 yılında Diffie ve Hellman ikilisi geleneksel kriptografi yaklaşımlarını alt-üst ederek yeni bir yaklaşım olan asimetrik şifreleme algoritmasını geliştirmişlerdir. Asimetrik şifreleme yaklaşımında biri açık/genel, diğeri gizli/özel

olmak üzere, şifre kullanıcılarının iki anahtarı mevcuttur. Bu özelliği itibariyle e-imza uygulamalarına uyumludur.

1.4.2 E-imza Oluşturmada Bütünlüğü Sağlayan Özetleme (Hash) Algoritmaları ile İlgili Yapılan Çalışmalar

E-imza ile imzalanmış bir metnin bütünlüğü özetleme fonksiyonlarıyla sağlanır. Hash (özetleme); verilen herhangi bir uzunluktaki metni sabit uzunluktaki özetinin elde edilmesini sağlarlar.

- SHA **Özetleme Fonksiyon Serisi**; Amerika'da NSA (National Security Agency) ve ilk olarak 1993 yılında FIPS PUB 180 standardında yayınlanmıştır. Devam eden yıllarda aynı seride SHA-0, SHA-1, SHA-2 (SHA-224, SHA-256, SHA-384 ve SHA-512 olmak üzere dört alt grubu mevcuttur.)[6]
- **RIPE-MD-160 Özetleme Fonksiyonu Serisi**; Bu algoritma Avrupa Birliği tarafından kullanılan bir algoritmadır.
- **MD Özetleme Fonksiyonu Serisi**; MD Serisi Ron Rivest tarafından geliştirirlen bir algoritmadır. Bu algoritma ailesindeki algoritmaların hepsi 128-bitlik özetleme sağlar.
- **MAC Özetleme Fonksiyonu Serisi** (Message Authentication Codes); Tek anahtarla mesaj oluşturma ve doğrulama şekline dayanan bir algoritmadır.

Özetleme fonksiyonları e-imzalı bir verinin bütünlüğünü sorgulamada ve iletim durumunda imzalama işlemi sonucu büyüyen veriyi sıkıştırmada kullanılır.

1.4.3 E-imza Oluşturmada İnkar Edilmezlik ve Kimlik Doğrulama ile İlgili Yapılan Çalışmalar

E-imza ile ilgili "İnkar Edilmezlik ve Kimlik doğrulama" alanında yapılan çalışmalar, devletlerin ve alt kurumlarının "Açık Anahtar Alt Yapısı" üzerinde kamusal ve ticari alanlarda gerçekleştirmiş olduğu e-imza çalışmaları/uygulamalarıdır.

1.5 Avrupa Birliğinde E-imza Çalışmaları

AB Konseyi 13 Aralık 1999'da 99/93/EC sayılı Elektronik İmza Direktifini kabul etmiştir. Bu çerçevede, üye ülkelerin Direktifi ulusal hukuklarına yansıtması

için konulan süre 19 Temmuz 2001 olarak belirlenmiştir. Söz konusu Direktif güvenlik ve sorumluluk ile ilgili asgari kurallar getirmekte, hizmetlerin serbest dolaşımı temelinde elektronik imzaların AB çapında hukuken tanınmasını sağlamayı amaçlamakta ve elektronik imzanın kullanılması ve hukuken tanınması için gereken çerçeveyi oluşturmaktadır. [7]

Avrupa Birliği'ne bağlı üye ülkelerin, e-imzayı kanun kabul etmeleri Dünya'daki e-imza çalışmaları ile birlikte tablo-1'de gösterilmiştir.

1.6 Türkiye'de E-imza Çalışmaları

5070 sayılı Elektronik İmza Kanunu 23.01.2004 tarihli ve 25355 sayılı Resmi Gazete'de yayımlanmış ve 23.07.2004 tarihinde yürürlüğe girmiştir. Ayrıca tüm kamu kurum ve kuruluşlarının aynı kurumsal sertifika yapısı altında toplanması için, Kamu Sertifikasyon Merkezinin kurulması ve işletilmesi görev ve sorumluluğu , Başbakanlık Türkiye Bilimsel ve Teknik Araştırma Kurumu'na bağlı Ulusal Elektronik ve Kriptoloji Araştırma Enstistüsü Müdürlüğü'ne (UEKAE) verilmiştir.[2]

Ulusal Elektronik ve Kriptoloji Araştırma Enstistüsü'nün kamu sertifika oluşturma görevi TÜBİTAK 'a verilmiştir. Sivil kurumlara sertifika oluşturmak için ise E-Güven, Turktrust ve E-Tuğra şirketleri 5070 Sayılı Elektronik Kanunu'nun 8.Maddesi'ne göre hizmet sunmaktadırlar.

1.7 Dünya'da E-imza Çalışmaları

E-imza uygulamalarının yaygınlaşması, e-imzanın uygulandığı taktirde**,** sağladığı iş ve işlemlerin gerçekleştirilme hızı ve kalitesinin anlaşılması noktasında başlamıştır. Dünya da çeşitli ülkelerde yapılan e-imza çalışmaları aşağıdaki tabloda (Tablo-1) görmek mümkündür.

Ülkeler	Yasalar	Tarihler
Malezya	Sayısal İmza Yasası	1998
Singapur	Elektronik İşlemler Yasaı	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	Elekronik Haberleşme Yasası	2000
İrlanda	Elektronik Ticaret Kanunu	2000
İsrail	Elektronik İmza Yasası	2000
Litvanya	Dijital İmzalar Yasası	2000
Lüksemburg	E-Ticaret 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

Hollanda	Sayısal İmza Kanunu	2001
Rusya	Dijital Elektronik İmzalar Federal Yasası	2002
Slovak Cum.	Elektronik İmza Yasası	2002
Türkiye	Elektronik Veri, Elektronik Sözleşme ve Elektronik İmza Yasası	2004

Tablo-1 Dünyada Elektronik İmza Yasaları Uygulama Yılları (Sağıroğlu,Alkan,2005)

İKİNCİ BÖLÜM
TEKNİK AÇISINDAN E-İMZA VE AÇIK ANAHTAR ALTYAPISI

2. E-İmza Oluşturma

Türkiye'de kabul edilen güvenli e-imza oluşturma standartları, Avrupa Birliği Direktifi'ne dayanarak hazırlanmıştır. Elektronik İmza Kanunu'nda "Güvenli Elektronik İmza" olarak isimlendirilen nitelikli elektronik imza şu özellikleri taşımalıdır:

- Sadece imza sahibine bağlı olmak
- İmza sahibinin kimliğini tespitini sağlamak
- Sadece imza sahibinin kontrolünde oluşturulmak
- İmzalanmış veride sonradan değişiklik yapılıp yapılmamış olduğunun tespitini sağlamak.

"Farklı amaçlara hizmet edebilecek şekilde tasarlanmış elektronik imza formatları, EESSI (European Electronic Signature Standardization Initiative) altında, ETSI (TS 101 733)[2]'de tanımlanmıştır. Yine bu kapsamda, imza ve imzada kullanılan sertifika doğrulama süreçleri, CEN/ISSS Workshop on Electronic Signatures yayını olan CWA 14171[3]'de tanımlanmış ve ayrıntılarıyla açıklanmıştır."[8]

2.1 Basit Elektronik İmza

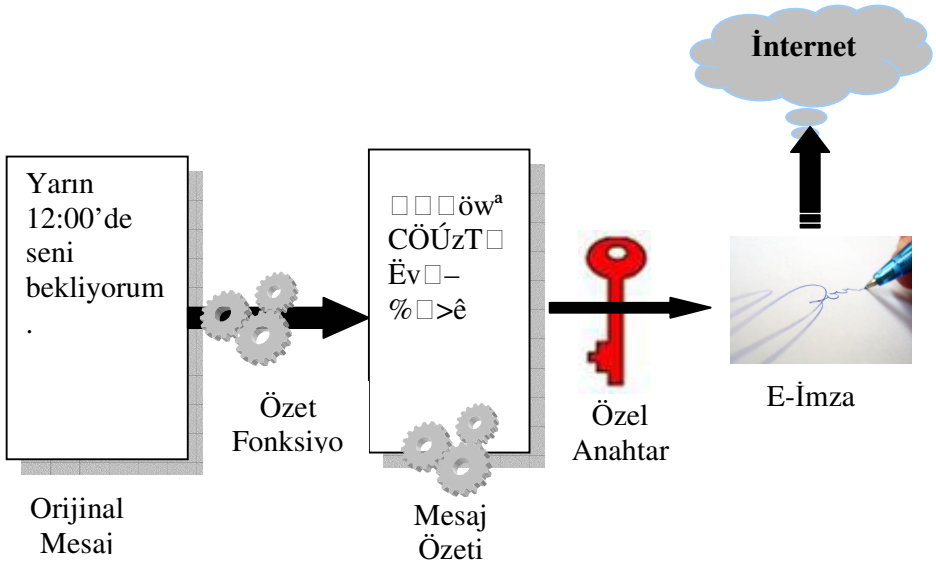
Elektronik imza formatlarının en basit ve minimum özelliklere sahip yapısıdır. Bu imza formatı, diğerlerine de temel teşkil etmekte olup imza yapısının iskeleti burada oluşturulmaktadır. İmzalı veri yapısı, RFC 3852 CMS (Cryptographic Message Syntax) standardında açıklanmıştır.

CMS yapısı, genel olarak veri korumasına yönelik elektronik imza, özet, şifreli mesaj yapılarını tanımlayan standarttır. Bu standarda göre oluşturulacak kriptografik mesaj yapısı, içerik tipi ve içerik bilgilerinden oluşmaktadır. İmzalı veri oluşturmak için izlenecek adımlar, Şekil 1'de de gösterildiği gibi:

- İmzalanacak verinin, kullanıcı sertifikasındaki özet algoritması kullanılarak mesaj özeti oluşturulur ,
- Oluşturulan özet imza sahibinin özel imzalama anahtarı ile imzalanır,
- Oluşturulan imzalı veriye imza sahibinin sertifika bilgisi eklenir ,
- İmzalı veri haberleşme ortamı üzerinden alıcıya gönderilir.

Basit Elektronik İmza formatında bulunması gereken nitelikler İçerik Tipi (content-type), Mesaj Özeti (message-digest) ve İmzalama Sertifikası (signing certificate) imzalı nitelikleridir. Bir CMS imza yapısının ETSI standardında tanımlanan Basit Elektronik İmza formatında olabilmesi için en azından bu nitelikleri taşıması gerekmektedir.[8]

Ayrıca e-imza ile imzalanmış verinin ağ üzerindeki dolaşımında, veri güvenliği sağlanmak istenirse, açık mesaj içeriğinin bir şifreleme algoritmasına tabi tutulması gerekir.



Şekil.1 Basit Elektronik İmza

Asimetrik şifreleme algoritmaları yavaş işlem kapasitesine sahip olduklarından, imzalı verinin güvenliğini sağlamada hızlı işlem yapma özelliğine sahip simetrik algoritmalar tercih edilebilir. Verinin imzalamadan önceki şifreleme yaklaşımı uygulamanın amacına göre tespit edilmesi daha doğru bir yaklaşımdır.

2.1.1 Çoklu İmza Algoritması

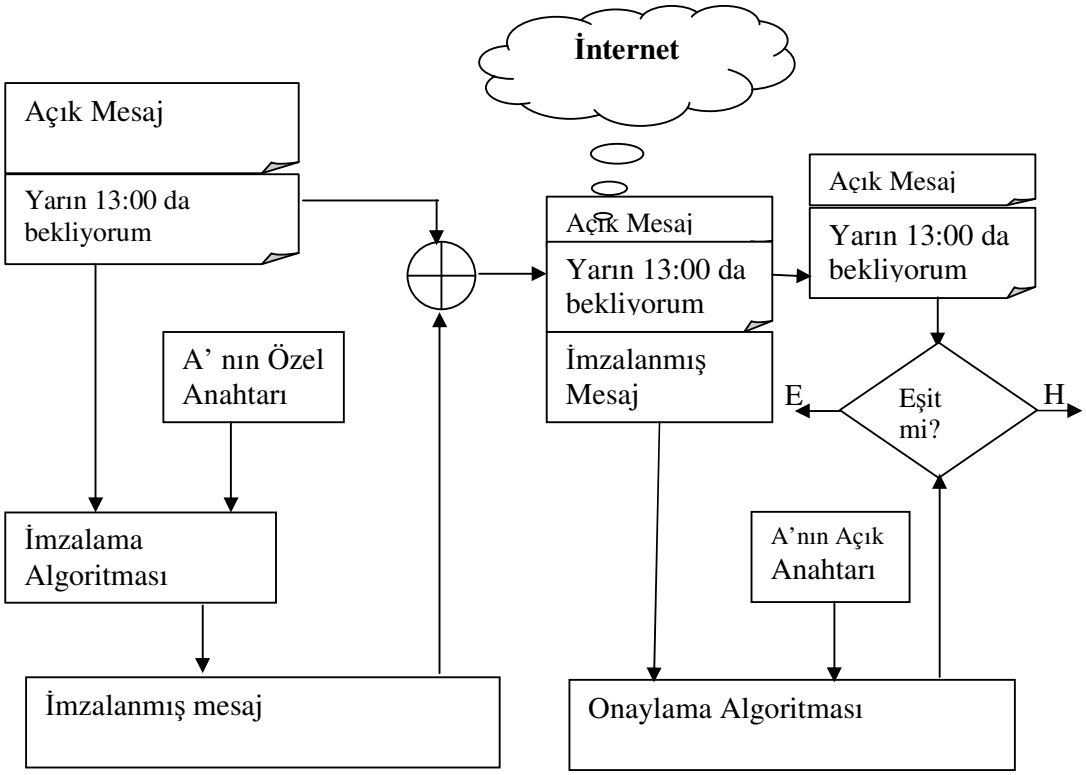
Bazı ticari ve kamusal uygulamalarda, aynı metne birden çok kişinin e-imzasının eklenme zorunluluğu doğabilir. Örneğin bir firmaya ait bir belgenin, şirketin genel müdürü, muhasebe müdürü ve depo sorumlusu tarafından imzalanması gerekiyorsa, bu imzalama işlemi basit e-imza formatında gerçekleştirilemez. Dolayısıyla tek anahtar çifti, aynı metin üzerinde kullanılması gereken e-imza uygulamaları için uygun değildir. Böyle bir durumda önerilen çözüm, Threshold kriptoloji sistemidir. Bu kriptoloji sistemi, özel anahtar bilgisinin parçalanarak dağıtılmasını öngörür. Böylece bu sistem ile çoklu imzalama uygulamaları gerçekleştirilebilir. [23]

2.1.2 E-imzada Kimlik Doğrulama

Bir Açık Anahtar Altyapısındaki, A isimli e-imza kullanıcısı, B isimli e-imza kullanıcısına bir imzalı mesaj göndersin. Gönderilen mesajın doğrulanmasının adımları aşağıda sıralandığı gibidir;

- Mesajı gönderen taraf olan A, göndereceği açık metni bir özet fonksiyonundan geçirir.
- A isimli e-imza kullanıcısı elde ettiği mesaj özetini kendi özel anahtarıyla imzalama algoritmasından geçirerek imzalanmış mesaj elde eder.
- Daha sonra orijinal açık mesaja imzalı özet mesaj ve A isimli e-imza kullanıcısının sertifika bilgileri eklenir.
- A isimli e-imza kullanıcısı elde ettiği imzalı mesajı B isimli e-imza kullanıcısına gönderir.
- B isimli e-imza kullanıcısına mesaj ulaştığında, A isimli e–imza kullanıcısının açık anahtarını kullanarak mesaj imzasını açar ve mesajın A isimli e–imza kullanıcısından gelip gelmediğini kontrol eder. A isimli e –imza kullanıcısının açık anahtarı mesaj imzasını açarsa, imzalı mesaj gerçekten A kullanıcısından gelmiştir.

- Mesajın A isimli e–imza kullanıcısından geldiği tespit edildikten sonra, onaylama algoritmasından elde edilen (A' nın gönderdiği özet) özet ile B isimli e-imza kullanıcısının açık metin olarak gelen orijinal metinden elde etmiş olduğu özet değeri karşılaştırılır (A ve B kullanıcıları aynı özetleme fonksiyonunu kullanmak zorundadır). Bu işlem sonunda elde edilen ve alınan özet değerleri eşit ise orijinal mesajın ağ üzerinde dolaşımdayken değişmediği kabul edilir.



Şekil-2 E-imzalama Süreci (Sağıroğlu, Alkan, 2005)

Böylece gelen mesaj üzerinden, hem "Kimlik Doğrulama" hem de "Mesaj Bütünlüğü" nitelikleri sorgulanmış oldu. Şekil-2 üzerinde söz konusu senaryolar gösterilmiştir.[2]

Bir e-imza kullanıcısının herhangi bir e-imzayı doğrulayabilmesi için bazı araçlara ihtiyacı vardır. İmzası doğrulanmak istenen e-imza kullanıcısının aşağıda belirlenen bilgileri ve e-imza oluşturma-doğrulama araçlarının nitelikleri belli olmalıdır:

- İmza sahibinin sertifikası ve sertifikasında kişisel bilgilerinin niteliği bilinmelidir.
- Elektronik sertifikanın hangi ESHS (Elektronik Sertifika Hizmet Sağlayıcı) dan aldığı bilgisi bulunmalıdır.
- E-imza kullanıcısının bağlı olduğu Açık Anahtar Alt Yapısı üzerinde faal çalışan Zaman Damgası ile zaman verisini oluşturan araçlar olmalıdır.
- Sertifika geçerlilik süresi e-imza doğrulama araçları üzerinden sorgulanabilmelidir.
- ESHS (Elektronik Sertifika Hizmet Sağlayıcı)' nin hangi Kök sertifika Makamına bağlı olduğunu sorgulanabilecek araçlar oluşturulmalıdır.

5070 Sayılı E-imza Kanununda e-imza oluşturma araçlarının teknik nitelikleri belirtilmiştir.

2.1.3 Güvenli E-İmza Doğrulama Araçları

- İmzanın doğrulanması için kullanılan verileri, değiştirmeksizin doğrulama yapan kişiye gösteren,
- İmza doğrulama işlemini güvenilir ve kessin bir biçimde çalıştıran ve doğrulama sonuçlarını değiştirmeksizin doğrulama yapan kişiye gösteren,
- Gerektiğinde, imzalanmış verinin güvenilir bir biçimde gösterilmesini sağlayan,
- İmzanın doğrulanması için kullanılan elektronik sertifikanın doğruluğunu ve geçerliliğini güvenilir bir biçimde tespit ederek sonuçlarını değiştirmeksizin doğrulama yapan kişiye gösteren,
- İmza sahibinin kimliğini değiştirmeksizin doğrulama yapan kişiye gösteren,
- İmzanın doğrulanması ile ilgili şartlara etki edecek değişikliklerin tespit edilmesini sağlayan, imza doğrulama araçlarıdır.][4]

2.2 Elektronik İmza Uygulamalarına İlişkin Standartlar

E-imza uygulamalarının bir standarda bağlı olması, Dünya da ülkemizde, gerek e-imza uygulamalarının entegrasyonu gerekse ortak güvenlik seviyesinin oluşturulması açısından oldukça önemlidir. Çünkü, e-imza uygulamalarının yürütüldüğü ortak alanlar üzerinde, standartlaşamama durumunun oluşturacağı uyumsuzluk, teknik ve hukuki açıdan da uyumsuzluğa neden olacaktır.

2.2.1 AB Ülkelerinde Kullanılan E- imza Standartları

- CWA 14167-1 (Mart 2003): Elektronik İmza Güvenilir Sistem Yönetim Sertifikaları için Güvenlik İhtiyaçları — Bölüm 1: Sistem Güvenlik İhtiyaçları
- CWA 14167-2 (Mart 2002): Elektronik İmza Güvenilir Sistem Yönetim Sertifikaları için Güvenlik İhtiyaçları — Bölüm 2: Sertifika Servis Sağlayıcıları İmzalama işlemleri için Kriptolama Modülü — Koruma Profili (MCSO-PP)
- CWA 14167-3 (Haziran 2003): Elektronik İmzalar için Güvenilir Sistem Yönetim Sertifikaları için Güvenlik İhtiyaçları – Bölüm 3: Sertifika Servis Sağlayıcıları Anahtar Oluşturma Hizmetleri için Kriptolama Modülü – Koruma Profili
- CWA 14169 (Mart 2002): nitelikli imza oluşturma araçları
- CWA 14355 (Haziran 2002): Güvenli İmza Uygulama Rehberi- Oluşturma Araçları[10]

Avrupa Birliği e-imza uygulamalarının etkinliği için teknik ve hukuksal açıdan bütün AB Ülkelerini kapsayan ortak standartlar oluşturarak, e-imza kullanımını yaygınlaştırmayı hedeflemiştir.

2.2.2 Türkiye'de 5070 Sayılı Kanuna Göre E-imza Oluşturmada Öngörülen Standartlar

Avrupa birliğine siyasal, sosyal, ekonomik ve kültürel açıdan bir katılım aşamasında olan ülkemizin, teknolojik olarak ta AB ile uyum sağlamak durumundadır. Bilişim teknolojilerindeki önemli bir gelişme olan e-imza uygulamalarının Türkiye'deki standartları, AB Standartları kıstas alınarak oluşturulmuştur.

Buna göre güvenlikle ilgili olarak aşağıda verilen standartların da göz önüne alınması gerekmektedir:

- TS ISO/IEC 17799 (2002) Bilgi teknolojisi — Bilgi güvenliği yönetimi için uygulama Prensipleri
- BS 7799-2 (2002) Bilgi Güvenliği Yönetim Sistemleri — Kullanım İçin Özellikler Rehberi

- TS ISO/IEC 15408-1 (2002) Bilgi teknolojisi — Güvenlik teknikleri- Bilgi teknolojisi (IT) güvenliği için değerlendirme kriterleri — Bölüm 1: Giriş ve genel model
- TS ISO/IEC 15408-2 (2002) Bilgi teknolojisi — Güvenlik teknikleri - Bilgi teknolojisi (IT) güvenliği için değerlendirme kriterleri — Bölüm 2: Güvenlik fonksiyonel gereksinimleri
- TS ISO/IEC 15408-3 (2003) Bilgi teknolojisi — Güvenlik teknikleri — Bilgi teknolojisi (IT) güvenliği için değerlendirme kriterleri — Bölüm 3: Güvenlik garanti gereksinimleri.[4]

2.2.3 Güvenli E-imza Oluşturma Araçları ile İlgili Standartlar

5070 sayılı Kanun'un 6.maddesinde güvenli elektronik imza oluşturma araçlarının aşağıdaki koşulları yerine getirmesi zorunluluğu belirtilmiştir:

- Ürettiği elektronik imza oluşturma verilerinin kendi aralarında bir eşi daha bulunmaması,
- Üzerinde kayıtlı olan elektronik imza oluşturma verilerinin araç dışına hiç bir biçimde çıkarılamamasını ve gizliliğini sağlaması,
- Üzerinde kayıtlı olan elektronik imza oluşturma verilerinin, üçüncü kişilerce elde edilememesi, kullanılamaması ve elektronik imzanın sahteciliğe karşı koruması,
- İmzalanacak verinin imza sahibi dışında değiştirilememesi ve bu verinin imza sahibi tarafından imzanın oluşturulmasından önce görülebilmesi.[4]

Yukarıda sayılan özelliklerden güvenli elektronik imza oluşturma aracı olarak akıllı kart ya da token ifade edilmiştir. İlgili donanımın özellikleri ve standartlarında BM UNCITRAL (United Nations Commission on International Trade Law-Birleşmiş Milletler Uluslararası Ticaret Hukuku Komisyonu) da belirtilen hususlar ve ilgili standartlar (CEN, ETSI, ISO vb.) değerlendirilmektedir .

E-imza oluşturulurken bir standarda bağlı kalma zorunluluğu, e-imza güvenliğinin sağlanmasındaki gerekliliktir. Bu nedenle, e-imza oluşturmada kullanılacak matematiksel algoritmalardan (kriptografik ve özet fonksiyonu oluşturma algoritmaları), e-imza yazılım ve donanımlarına kadar, bütün e-imza

bileşenlerinin kümelendiği "Açık Anahtar Altyapısı" nın bir standarda göre yapılandırılması şarttır.

Avrupa Konseyi ve Avrupa Parlamentosunun 13 Aralık 1999 tarih ve 1999/93/EC sayılı direktifi ile elektronik imza ile ilgili esaslar belirlenmiştir. Ek II - f maddesinde nitelikli sertifika verecek olan sertifika hizmet sağlayıcılarının değişikliklere karşı korunmuş, destekledikleri işlemlerin teknik ve kriptolojik güvenliği garanti edilen güvenilir sistemler ve ürünler kullanmaları şart koşulmuştur. Ayrıca burada yer alan Ek III'e göre, nitelikli imza oluşturma araçları aşağıdaki gereksinimleri sağlamalıdır:

1. Nitelikli imza oluşturma araçları, aşağıdaki koşulları uygun teknik ve yordamsal yöntemlerle garanti etmelidir:

a) İmza oluşturmak için kullanılan imza oluşturma verisi, pratikte sadece bir kez olabilir ve gizliliği rasyonel olarak garanti edilir.

b) İmza oluşturmak için kullanılan imza oluşturma verisine, rasyonel olarak ulaşılamayacağı garanti edilmelidir ve imza mevcut teknolojileri kullanarak sahtekarlığa karşı korunmalıdır

c) İmza oluşturmak için kullanılan imza oluşturma verisi, başkalarının kullanımına karşı yasal imza sahibi tarafından güvenilir bir şekilde korunabilmelidir.

2. Nitelikli imza oluşturma araçları imzalanacak veriyi değiştirememeli ya da imzalama süreci öncesi imza sahibine verinin sunulmasını önleyememelidir.

"Avrupa Elektronik İmza Standardizasyonu, ICT Standart Yönetimi (The ICT Standards Board, CEN, ETSI ve CENELEC Avrupa standart organizasyonlarının işbirliğinde oluşturulan bir inisiyatiftir) tarafından Avrupa Komisyonu direktifleri sonucunda ele alınmıştır. Elektronik imza oluşturma araçlarının gerek işlevi ve gerekse güvenlik kalitesi açısından standartlara uygun olması gerekir."[9]

Güvenli imza geliştirme araçlarının;

- Pratikte sadece bir kez üretilebilirliğe
- Gizliliğe,

- Yüksek kalitede anahtar üretimine ve anahtar korumasına,
- Güçlü algoritmalara ve yeterli anahtar uzunluğuna,
- PIN/şifre yapısının sözlük ve etkin bitirici ataklara dayanıklı olmasına,
- İmzanın üretildiği araç içerisinden anahtarın hiçbir zaman ayrılamama özelliğine,
- Aracın hiçbir yedek yada kopya üretilmeme özelliğine sahip olması gerekmektedir.[9]

2.2.3.1 E-imza Oluşturma Araçları

a. **Akıllı Kartlar**: E-oluşturmada önemli bir donanım bileşenidir. E-imza sahibinin imzayı oluşturmak için kullanmak zorunda olduğu ilk donanımdır. Ayrıca kullanıcısının e-imza bilgilerini içerdiği için de son derece iyileştirilmiş güvenlik koşullarında saklanmalıdır. Akıllı kartların temel özellikleri aşağıdaki gibidir;

- Akıllı kartlar bireylerin özel anahtarlarını güvenli şekilde tutarlar.
- Dijital imzalar yaratırlar.
- Akıllı kartlar entegre yongayı çalıştırmak için yerel tanılama (örneğin PIN, biometrik) sunarlar.
- Çoklu uygulama özelliklerini desteklerler.
- Akıllı kartların güvenlik seviyesi EAL-4+ derecesinde olmalıdır.
- Akıllı kartı sahibi dışında, kesinlikle kimsenin kullanımına sunulmamalıdır.

b) **Akıllı Çubuklar**: Akıllı çubuklar da tıpkı akıllı kartlar gibi e-imzanın oluşturulması ve kullanılması için kullanılan bir donanımdır.

- Kendilerine has işletim sistemlerine sahiptirler.
- İşletim sistemlerinin güvenlik derecesi EAL-4+ seviyesindedir.
- Hem akıllı kartlarda, hem de akıllı çubuklarda özel anahtar, donanım dışına kesinlikle taşınmamalı.
- Akıllı çubuklarda güvenlik seviyesinin arttırılmasına yönelik olarak, şifre girilmesi uygun bir çözümdür.

2.2.3.2 E-imza Oluşturma Yazılımları

Gerek Türkiye'de ve gerekse Dünya'da bir çok elektronik sertifika hizmet sağlayıcısı, yeterli seviyede artırılmış güvenlikli yazılımlar kullanmaktadır. Burada

sadece en çok kullanılan, ülkemizde ve Dünya'da bilinen birkaç yazılımı vermekle yetineceğiz. E-imza Yazılımlarının tercihi kişisel kullanım gereklerine ve ESHS (Elektronik Sertifika Hizmet Sağlayıcı) nın hizmetine ve güvenlik kalitesine göre seçilir. Günümüzde kullanılan e-imza yazılımları;

- ESYA: TÜBİTAK bünyesindeki UEKAE Tarafından geliştirilmiştir.
- Zeugma: Yine Türkiye'de TÜBİTAK'ın bir alt yazılım kurumu olarak çalışan BİLTEN tarafından geliştirilen bir yazılımdır.
- VeriSign: Amerika'da geliştirilmiş ve dünyanın saygın şirketleri tarafından kullanılan, AAA (Açık Anahtar Altyapısı) destekli bir yazılımdır.
- IBM Trust Authority: IBM tarafında geliştirilen bir yazılımdır.
- RSA Keon: Amerika'da RSA Security firması tarafından geliştirilmiş bir yazılımdır.
- Entrust/PKI: Amerika'da Entrust firması tarafından geliştirilmiş bir yazılımdır.
- Baltimore UniCERT: Amerika'da Baltimore Technology firması tarafından geliştirilmiş bir yazılımdır.[2]

2.3 E-imzada Zaman Damgası

Zaman damgaları, e-imzanın oluşturulma zamanını belirlemiş olup, imzalı veriyi kırılması olası özet algoritmalarına ve kriptografik algoritmaların kırılmasına karşı korumaktadır. E-imzada zaman damgasının veri üzerindeki zaman ibaresinin varlığı, veriyi alan ve veren açısından, inkar edilmezlik özelliğini pekiştiriyor. Özellikle hukuksal uyuşmazlıklarda, imzalı verinin imzalanma zamanı çok önemlidir. Açık Anahtar Alt yapılarında e-imza için zaman damgasını atomik saatler sağlar.

2.4 Açık Anahtar Alt Yapı (AAA)

Açık Anahtar Alt Yapı **(AAA)** asimetrik kriptografik yaklaşımlar üzerine inşa edilmiş olup, asimetrik kriptografinin özelliği gereği, Açık Anahtar Alt Yapı sisteminde kullanıcı durumda olan herkesin biri açık (genel/public), diğeri gizli (özel/private) olmak üzere iki tane anahtarı mevcuttur. AAA sisteminde gerçekleştirilen şifreleme ve şifre çözme işlemleri bu ikili anahtarlarla (açık ve gizli

anahtarlar) gerçekleştirildiğinde, bu yaklaşım Açık Anahtar Alt Yapı (AAA) olarak adlandırılmıştır.

Bilindiği gibi AAA' nın üzerine inşa edildiği asimetrik şifreleme yaklaşımı geliştirilmeden önce, geleneksel olarak sadece tek gizli anahtarın bilinmesini zorunlu kılan simetrik şifreleme yöntemi kullanılıyordu. Simetrik şifreleme yönteminde anahtar dağıtımı başta olmak üzere, birçok sakıncaya sebep olmakta ve tamamen gizlilik ilkesine dayandığından, toplumun genel kullanımına sunulabilecek işlevi olmamaktaydı.

AAA elektronik ortamlarda bilginin dolaşımı sırasında, güvenliğini, inkar edilmezliğini, bütünlüğünü, kimlik doğrulama ve inkar edememe gibi işleyişleri gerçekleştirebilmektedir. Dikkat edilirse bu özellikler e-imzanın özellikleridir. Dolayısıyla AAA şifreleme yaklaşımı ve bu yaklaşıma uygun geliştirilen protokoller/standartlar, bu protokol/standartlara göre oluşturulan ve güvenliği üst seviyede (hükümetler seviyesinde) sağlanarak oluşturulan kurumlar ve e-imzanın teknik ve hukuksal bütün özellikleri ile kullanımını sağlayan yapıdır. Kısaca söylemek gerekirse AAA yaklaşımı dışında e-imza kullanımı gerçekleşememektedir. En azından kriptografi biliminin bugünkü gelişim aşamasında, varolan yaklaşımlarla, AAA 'ya henüz alternatif bir yaklaşım yoktur.

2.4.1 AAA'nın Oluşturulması

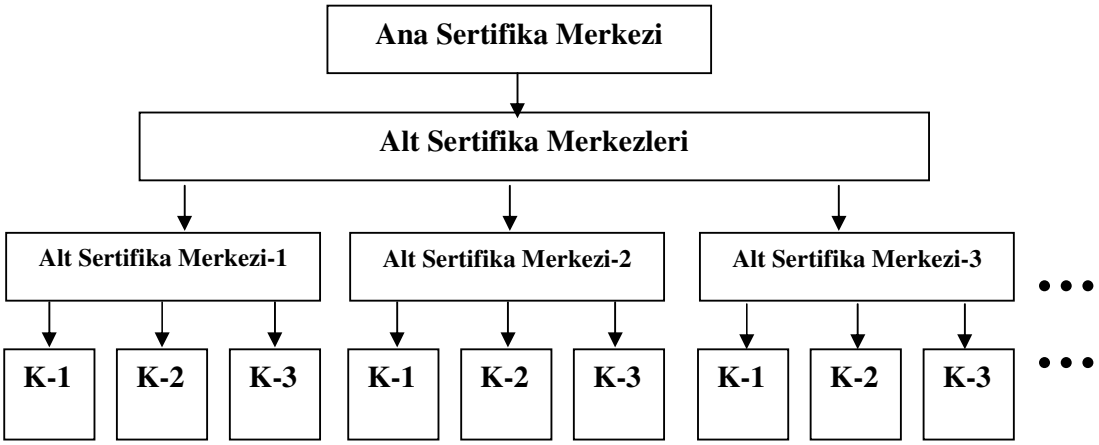
Öncelikle belirtelim ki, daha önce anlatmaya çalıştığımız e-imza ile ilgili standartlar, kriptografik yaklaşımlar (asimetrik kriptografi), imzalama ve imza doğrulama araçları, e-imza yazılımları ve e-imza ile ilgili işlerlik kazanan kurumların hepsi AAA'nın birer alt bileşenidir.Bir AAA ;

- AAA'nın inşa edildiği ülkede hükümet seviyesinde sağlanmış güvenlik derecesiyle, bütün sertifika merkezlerinin bağlı olduğu "Ana Sertifika Merkezi " oluşturulmalı.
- Ana Sertifika Merkezine bağlı, kamu ve özel sektöre hizmet veren "Alt Sertifika Merkezleri " oluşturulmalı.

- Ana Sertifika Merkezi ile Alt Sertifika Merkezlerinin kullandığı yazılım ve donanımlar, e-imzanın güvenli kullanımını sağlayacak düzeyde olmalı ve yeniliklerin/değişimlerin uyarlanabilecekleri yapıda olmalılar.

- Ana Sertifika Merkezi ile Alt Sertifika Merkezlerinin fiziksel güvenliği sağlanmalı ve bu merkezlerde çalışan personelin istenen nitelikte olmasına dikkat edilmelidir. (Sertifika merkezlerinde çalışacak personelin taşıması gereken nitelikler 5070 Sayılı Elektronik İmza Kanunun'da belirtilmiştir.)
- Sertifika merkezleri, e-imza kullanıcılarına olası mağduriyetlere karşı, belirlenen bir oranda sigorta teminatı sağlamalı.
- AAA'nın icra edildiği ülkenin hukuksal normlarına uygun olarak oluşturulmalı ve AAA yapısı üzerinden gerçeklenen işlemlerin hukuksal geçerlilik sorunu çözülmeli.
- AAA yazılımları kullanıcının çok rahat kullanabileceği şekilde oluşturulmalı.[20]

Türkiye'de Ana Sertifika Merkezi Türk Telekomünikasyon kurumudur. Alt Sertifika Merkezleri olarak da Türktrust, E-Tuğra ve E-Güven sertifika hizmet sağlayıcıları, e-imza çalışmalarını sürdürmektedirler. Kamuya ait sertifikaların teminini TÜBİTAK-UEKAE üstlenmiş durumdadır. Bir AAA yapısının nasıl olması gerektiği ile ilgili standartlar ve teknik nitelikler 5070 Sayılı Elektronik İmza Kanunun' da belirtilmiştir.



Şekil – 3 AAA' nın Genel Yapısı

2.5 AAA Yapısında ESHS (Elektronik Sertifika Hizmet Sağlayıcı)

Elektronik Sertifika Hizmet Sağlayıcıları e-imza ile ilgili yazılım ve donanımları sağlayan, kişiyi e-imzayı kullanması konusunda yetkili kılan, sağladığı sertifika ile AAA yapısında tanımlayan bir otoritedir. ESHS, Ana Sertifika Merkezi ile kullanıcı arasındaki bir kurum olup, e-imzanın geçerlilik durumunu sağlayan ve e-imzanın kullanıldığı bütün işleri kayıt altına alan, önemli bir omurgadır.

5070 Sayılı Elektronik İmza Kanunun'da elektronik sertifika hizmet sağlayıcısı; elektronik sertifika, zaman damgası ve elektronik imzalarla ilgili hizmetleri sağlayan kamu kurum ve kuruluşları ile gerçek veya özel hukuk tüzel kişilerdir. Elektronik sertifika hizmet sağlayıcısı yapacağı bildirimde;

- Güvenli ürün ve sistemleri kullanmak,
- Hizmeti güvenilir bir biçimde yürütmek,
- Sertifikaların taklit ve tahrif edilmesini önlemekle ilgili her türlü tedbiri almak, ile ilgili şartları sağladığını ayrıntılı bir biçimde gösterir,[10] şeklinde, nitelikleri ile beraber tanımlanmaktadır.

Elektronik sertifika hizmet sağlayıcıları sertifika iptallerini gerçekleştirir, bağlı olduğu AAA yapısı içerisinde kullanıcıların açık anahtarını yayımlar, açık ve gizli anahtarların elde edilmesindeki gizliliği sağlar. Şekil-3 te ifade edilen Alt Sertifika Merkezleri işleyiş, nitelik ve yetkileri itibariyle birer elektronik sertifika hizmet sağlayıcısıdırlar.

ESHS ' nin e-imza kullanıcısına verdiği sertifika, 5070 Sayılı Elektronik İmza Kanunun' da Nitelikli Elektronik Sertifika olarak aşağıdaki özellikleri ile ifade edilmiştir. Sertifikanın nitelikli olması, e-imza uygulamalarında kullanıcı açısından çok önemlidir. Çünkü kullanılan basit sertifikalarla gerçekleştirilen imzalama olaylarında problem çıkabilir. Elektronik sertifika, e-imza kullanıcısının elektronik ortamdaki kimlik kartıdır. 5070 Sayılı Elektronik İmza Kanunun' a göre elektronik sertifika;

"İmza sahibinin, imza doğrulama verisini ve kimlik bilgilerini birbirine bağlayan elektronik kaydıdır."[10]

Nitelikli elektronik sertifikada;

- Sertifikanın "nitelikli elektronik sertifika" olduğuna dair 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,
- Sertifika geçerlilik süresinin başlangıç ve bitiş tarihlerinin,
- Sertifika seri numarasının,
- Sertifika sahibi diğer kişi adına hareket ediyorsa bu yetkisine ilişkin bilginin,
- Varsa sertifikanın kullanım şartları ve kullanılacağı işlemlerdeki maddi sınırlamalara ilişkin bilgilerin,
- Sertifika hizmet sağlayıcısının sertifikada yer alan bilgileri doğrulayan güvenli elektronik imzanın, bulunması zorunludur.[10]

ELEKTRONİK SERTİFİKA	
Sertifikanın Seri Numarası:	
Sertifika Sahibinin Kimlik Bilgileri:	
Sertifikanın geçerlilik tarihi:	
Sertifikanın geçerlilik Süresi:	
Kullanılacak algoritmalar	
Açık Anahtar bilgisi:	
Sertifikayı Yayınlayan ESHS:	
Sertifikayı Yayınlayan ESHS' nin Elektronik İmzası:	

Şekil-4 Elektronik Sertifika

2.6 AAA Uygulamalarında Karşılaşılan Sorunlar

Bir AAA yapısının;yazılım,donanım, personel kalitesi ve fiziksel saldırılar vb. çok çeşitli risk ve saldırı durumları vardır. Ayrıca doğal afetlerle olabilecek tahribatlar için çözüm önerileri geliştirilmeli, sertifika hizmet sağlayıcılarının

güvenlilik sorgusu en üst düzeyde yapılmalı.Kayıp ve iptal edilmiş sertifikalarla ilgili ivedi bilgi sahibi olunmalıdır. Sertifikaların ikinci kez elde edilmesinde karşılaşılabilecek sorunlar çözülmelidir. AAA'ya duyulan güven e-imzanın işleyişinin temel garantisidir.

Bir AAA yapısında oluştururken çıkması muhtemel sorunlar aşağıdaki başlıklarla ifade edilebilir:

a. Mahremiyetin Korunamaması

"SSL bağlantıları sırasında bazı sunucular istemci sertifikasını istemekte, istemci tarafında çalışan Internet tarayıcı programlar da bu sertifikaları sunucuya otomatik olarak göndermektedir. Böylelikle sertifika sahibinin kimliği ve e-posta adresi gibi bazı bilgileri sertifika ile beraber kontrolsüz şekilde Internet üzerinde dolaşmış olmaktadır. Bu durum mahremiyet savunucularının sertifika sistemlerine karsı en önemli saldırısını oluşturmaktadır."[20]

b. Kayıt Zorlukları

Herhangi bir ESHS kurumundan sertifika edinmek isteyen kişilerin kişisel bilgilerinin bazı durumlarda gerçeğine uygun olmama ihtimali, sertifikanın hukuksal niteliğini (geçerliliğini) zedelemektedir.[2]

c. Sertifika Alımının Ücretli Olması

ESHS kurumlarından edinilen sertifikaların ücretli veya imza kullanıcısına ağır maliyetler getirmesi, e-imza kullanıcısının çoğalmasını engellemektedir.[2],[20]

d. Sertifika Otoritesine Karşı Güvensizlik Sorunu

Sertifika oluşturma yazılımlarının kolay olması, kötü niyetli kişilerin kendi amaçları doğrultusunda, bu yazılımlarla yaratacakları sertifikaları gelişigüzel dağıtma eğilimleri her zaman bir tehlike teşkil etmektedir. Ana Sertifika otoritesine bağlı ESHS ların kullanıcı tarafından sorgulanma imkanı ile bu tür kötü niyetli yazımların AAA yapısı içerisinde barınma şansını yok etse bile , bilinçsiz kullanıcılar bazında önemli bir tehlikedir.

"SSL ve S/MIME sertifikalarını doğrulamak için gerekli kök SO (Sertifika Otoritesi) sertifikalar Internet tarayıcı programlarla beraber gelmektedir. Kullanıcılar bu kök SO'lara güvenmek zorunda bırakılmaktadır."[20]

Bu sorunlar ve benzer sorunlar AAA için ortaya çıkabilecek sorunlar olarak öngörülmektedir. Bilişim teknolojilerinin ve kriptoloji biliminin gelişim seyrine paralel oluşturulan güvenlik algoritmaları ve güvenlik protokolleri güvenlik seviyesini arttırırken, buna karşın kötü niyetli kişiler, bu güvenlik algoritmaları ve güvenlik protokollerinin açıkları üzerinden yeni saldırılar planlayacaklardır. Bu güvenlik evriminin yaşam döngüsü yüzde yüz güvenlik olamayacağının sonucuna götürür.

2. 6.1 Açık Anahtar Dağıtım Sorunları

Daha önce de belirttiğimiz gibi Açık anahtar Alt yapılı algoritmalarda iki anahtar vardır. Bunlar, birincisi şifrelemede ve imza doğrulamada kullanılan ve herkesin bildiği açık anahtar (public), ikincisi ise, deşifrelemede ve imza atmakta kullanılan ve sadece sahibinin bildiği gizli (private) anahtardır. Bu anahtarlar arasında matematiksel bağıntı olup, bilinen açık anahtardan, gizli anahtar elde etmek mümkün değildir denilebilir (yada çok çok zordur.)

Haklı olarak bu durumun anahtar dağıtım sorununu çözdüğünü sonucuna varırız. Çünkü simetrik şifreleme algoritmalarına göre, asimetrik şifreleme sistemleri anahtar dağıtımını çok ileri bir seviyede ve kolay bir mantıkla çözmüş durumdadır. Bilindiği gibi simetrik şifreleme yaklaşımı tek gizli anahtarın herkesçe paylaştırılması ilkesine dayanıyordu. Oysa daha önce hiç karşılaşmadığımız bir AAA yapısındaki birinin sadece asimetrik şifreleme ile belirlenmiş açık anahtarını bilmek, ona şifreli bir mesaj göndermek için yeterlidir.

Hemen belirtelim ki bu durum anahtar dağıtım problemini tamamen çözememiştir. Açık (public) olarak beyan edilen açık anahtarların gerçekte kime ait olduğu konusunda resmi düzeylerde sağlanmış güvencelerle ikna olmalıyız. Bu konuda çözüm; açık anahtar sahibinin imzasını oluşturmak ve kendisine gelen imzalı mesajı doğrulamak için kullandığı sertifikasının, o ülkedeki "Sertifika Otoritesi" tarafından verilmiş olması ile sağlanır. Çünkü resmi düzeyde Sertifika Otoritelerince verilmeyen, sadece kişisel beyanlara dayalı sertifikaların hukuksal geçerliliği yoktur.[20]

ÜÇÜNCÜ BÖLÜM
KRİPTOGRAFİK AÇIDAN E-İMZA

3. Genel Olarak Şifreleme (Kriptografi)

Kriptoloji bilimi kavramsal olarak Yunanca'da gizli dünya anlamına gelen "kryptos logos" kelimelerinden türetilmiş olup, insanlığın başlangıcından bu yana süregelen devamlılığı kabul edilen, günümüzde ise matematik, optik, bilgisayar gibi birçok alt bilim dalından beslenen ve bilgi güvenliğini amaçlayan bir bilimdir. Başka bir tanımlama ile kriptografi; bir belgenin veya verinin bir yerden başka bir (herhangi bir yolla) gönderilirken, belge içeriğinin, matematiksel fonksiyonlar yardımıyla gerçek formundan, üçüncü şahıslarca bilinmeyen bir başka forma dönüştürülmesi işlemidir.

Kriptolojide veri şifreleme/encryption, şifreli veriyi çözüp, gerçek metni elde etmeye ise deşifreleme/decryption denir. Kriptografide bir başka önemli kavram ise kriptanalizdir. Kriptanaliz; kriptografik sistemleri, bu sistemleri sağlayan matematik, optik, elektronik ve diğer şifre oluşturma yöntemlerini ve bu yöntemlere ait algoritmaları inceleyen ve şifreli verileri çözmeye çalışan yaklaşımları kapsar.

Bilişim sektörünün çok ileri düzeylerde geliştiği ve hayatın bir çok kullanım alanına yayıldığı günümüzde, özellikle bilgisayar sistemleri ve bu sistemler arasında dolaşan bilgilerin güvenliği daha çok önem kazanmakta ve bu durum devam ettikçe, kriptografi bilimi de o oranda önem kazanacaktır.

3.1 Basit (Klasik) Kriptografi ve Matematiksel İfadesi

Kriptografinin esas konusu iki kişi arasındaki herhangi bilginin güvenli olamayan bir kanalla paylaşılmasıdır. Bu kişiler Alice, Bob ve Oscar olarak kabul

edilir. Burada Alice mesajı gönderen, Bob mesajı alan ve Oscar ise mesajı gönderilen kanal üzerinde dinleyen (saldırgan) olarak bilinir. Alice güvenli olmayan bir kanal (bilgisayar ağı veya telefon hattı) üzerinden orijinal mesajı, Bob'la daha önce anlaştıkları gizli bir anahtarla şifreleyerek Bob'a gönderir. Oscar iletim kanalı üzerinde şifreli mesajı elde etmesine rağmen, gizli şifreleme anahtarını bilmediğinden orijinal metni elde edemez. Fakat orijinal mesaj Bob'a ulaştığında, Bob daha önce bildiği gizli şifreleme anahtarıyla, şifreli mesajı deşifre ederek orijinal mesajı elde eder. Bu konsept genel olarak aşağıdaki matematiksel notasyonla ifade edilir.

Tanım:Söz konusu olan kriptografi sistemi beş elemanlı sonlu bir küme üzerinden P, C, K, E, D tanımlanabilir.

1. P sınırlı büyüklükte bir metindir.

2. C sınırlı büyüklükteki şifreli metindir.

3. K anahtarlar kümesini temsil eder.

4. Her bir $K \in \kappa$ için $e_K \in \varepsilon$ şifreleme kuralı; ve uygun deşifreleme yöntemi, $d_K \in D$'dir. Şifreleme işlemi $e_K : P \to C$ ve deşifreleme işlemi ise $d_K : C \to P$ algoritmasına göre gerçekleştirilir. Burada şifreleme ve deşifreleme için tanımlanan genel fonksiyon ; $d_K(e_K(x)) = x$ olup, bütün metinler için, $x \in P$ dir.

Burada ana kural dördüncü kuraldır; x olarak kabul edilen bir mesaj içeriğinin, e_K ile şifrelendiğini ve sonrasında şifreli mesajın d_K ile deşifre edilip orijinal mesajın elde edildiğini ifade eder.

Alice ve Bob belirledikleri bir şifreleme sisteminin protokolünü uygularlar. Bunun için önce bir gizli anahtar seçerler; Anahtar seçimi sırasında Alice ve Bob'un aynı yerde olduklarını veya bu gizli anahtar seçimini farklı yerlerde olma durumunda güvenli bir yolla paylaştıklarını kabul ediyoruz. Alice Bob'a güvenli olmayan bir kanal üzerinden olan bir mesajı göndermiş olsun. İçeriği bir dizi (string) olan mesajı;

$$x = x_1 x_2 \cdots x_n$$

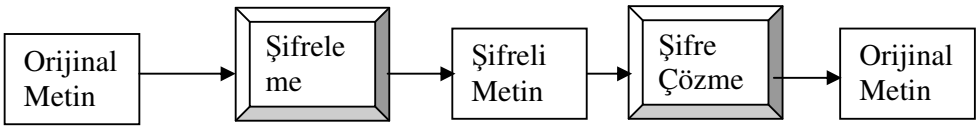
şeklinde ifade edebiliriz. Bazı tamsayılar için $n \geq 1,$ olmak üzere, $1 \leq i \leq n$

aralığında içeriği dizi olarak kabul edilen mesaj; $x_i \in P$ olarak temsil edilebilir. Her x_i daha önce belirlenen gizli K anahtarı kullanılarak, e_K şifreleme fonksiyonu ile şifrelenir. Böylece Alice $1 \le i \le n$ aralığında, $y_i = e_K(x_i)$ değerini hesaplayarak şifreli mesajı elde etmiş olur. Elde edilen;

$$y = y_1 y_2 \cdots y_n$$

şifreli dizisi kullanılacak kanal üzerinden Bob'a gönderilir. Bob kendisine gelen şifreli mesaj dizisini d_K şifre çözme fonksiyonu ile orijinal mesajı elde eder.[12]

Şekil-5 iletişim kanalındaki bir mesajın şifreleme ve deşifreleme akışını göstermektedir.



Şekil-5 Genel Olarak Şifreleme

Şunu belirtelim ki; bir e_K şifreleme fonksiyonu mutlaka tersinir (injecktive) bir fonksiyon olmalıdır. Bilindiği gibi tersinir fonksiyonlar, adından anlaşılacağı gibi tersi alınabilen, bire-bir ve örten fonksiyonlardır. Tersinir bir fonksiyonla şifrelediğimiz mesajı, aynı fonksiyonun tersini şifreli mesaja uygulayarak orijinal mesajı elde edebiliriz.

Kriptografik yöntemler/algoritmalar, genel olarak kullandıkları anahtarın niteliğine bağlı olarak ikiye ayrılırlar. Bu yöntemler simetrik ve asimetrik şifreleme algoritmalarıdır.

3.2 Şifreleme Algoritmalarının Performans Kriterleri

Kullanılan bütün teknolojik araç, gereç, yazılım, donanım ve algoritmik yaklaşımların sahip olması gereken kalite ve kullanılabilirlilik standardı, şifreleme algoritmaları için de geçerlidir. Avrupa ve Amerika Birleşik Devletlerin'de

şifreleme standartlarını belirleyen laboratuvarlar faaliyetlerini sürdürmektedirler. Bir şifreleme algoritmasının kalite ve kullanılabilirlik standartı;

- Kırılabilme süresinin uzunluğu.
- Şifreleme ve şifre çözme işlemlerine harcanan zaman. Zaman Karmaşıklığı.
- Şifreleme ve şifre çözme işlemlerinde ihtiyaç duyulan bellek miktarı (Bellek Karmaşıklığı).
- Bu algoritmaya dayalı şifreleme uygulamalarının esnekliği.
- Bu uygulamaların dağıtımındaki kolaylık yada algoritmaların standart hale getirilebilmesi.
- Algoritmanın kurulacak sisteme uygunluğu[11] olarak sıralanabilir.

Kriptografik yöntemler/algoritmalar, genel olarak kullandıkları anahtarın niteliğine bağlı olarak ikiye ayrılırlar. Bu yöntemler simetrik ve asimetrik şifreleme algoritmalarıdır. Kullanım amacı esas olmak şartıyla, şifreleme algoritmasının simetrik veya asimetrik olması özelliği de kesinlikle, şifreleme standardında bulunması gereken özelliklerin başında gelir.

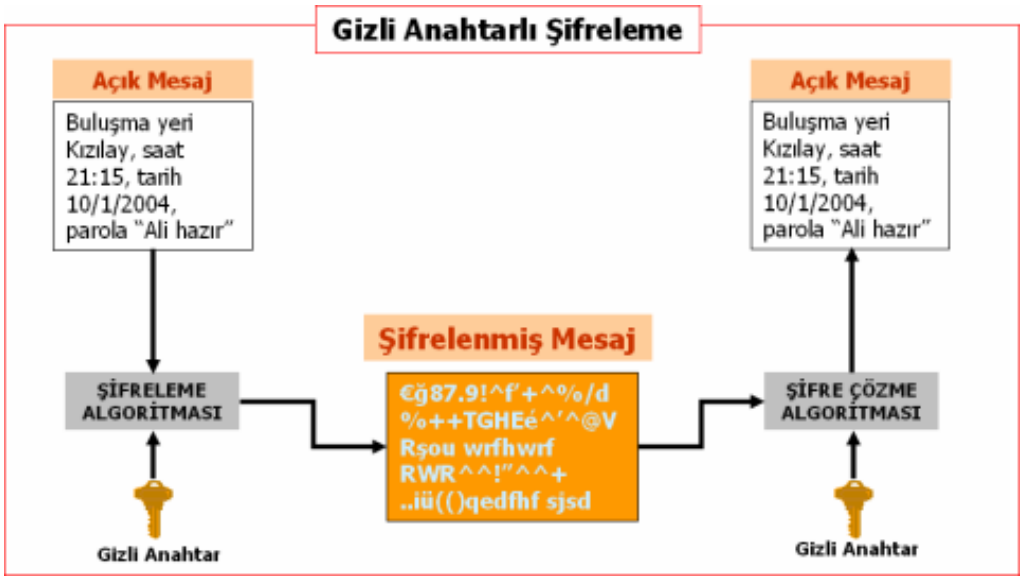
3.3 Simetrik Şifreleme Algoritmaları

Simetrik şifreleme algoritmalarında şifreleme ve deşifreleme işlemleri için tek bir gizli anahtar kullanılmaktadır. Simetrik şifreleme yaklaşımında, şifreleme işlemlerini gerçekleştirdikten sonra şifreli metni alıcıya gönderilir. Eğer mesajı alacak olan kişi daha önce gizli anahtarı bilmiyorsa, mesajın gizli anahtarının da alıcıya güvenli bir şekilde gönderilmesi gerekmektedir.

Simetrik şifreleme yaklaşımı ile Alice Bob'a bir şifreli mesaj göndermek isterse, aşağıdaki senaryo izlenecek:

- Alice ve Bob şifreleme için kullanacakları bir kriptogarfi sistemi üzerinde anlaşma sağlarlar.
- Daha sonra ortak belirleyecekleri bir anahtar seçerler.
- Alice seçilen kriptografik sistem üzerinden Bob'a göndereceği mesajı, daha önce belirlenen gizli anahtarla şifreleyerek, şifreli mesajı elde ederek gönderir.

- Bob belirlenen ortak kriptografik sistem üzerinden, daha önce seçilen gizli anahtarla mesajı deşifre ederek orijinal mesajı elde eder.[12]



Şekil-6 Simetrik şifreleme (Erol,H.-Aralık 2004)

Genel olarak şifreleme algoritmaları Blok şifreleme algoritmaları ve Bit katarı (dizi) şifreleme algoritmaları olmak üzere ikiye ayrılırlar:

- Blok şifreleme algoritmaları:AES, DES, IDEA, Skipjack, RC5
- Bit katarı (dizi) şifreleme algoritmaları:RC2, RC4.

3.3.1 Simetrik Şifreleme Algoritmalarının Dezavantajları

Simetrik şifreleme yönteminde temel şifreleme algoritması, gizli anahtar yöntemi ile gerçekleştiğinden, güvenli mesajlaşmanın sağlanması, yine güvenli bir anahtar dağıtımı ile mümkün olmaktadır. Sınırlı kişilerin kullandığı bir iletişim kanalında anahtar dağılımı problem olmamaktadır. Yalnız iletişim ağındaki kişi sayısı artıkça anahtar dağıtımı çok büyük sorunlar doğurmaktadır. Simetrik şifrelemede ***n*** iletişim kanalındaki kişi sayısı olmak üzere toplamda; ***n(n-1)/2*** adet anahtar gerekmektedir. Örneğin 100 adet çalışanı olan bir şirketin simetrik şifreleme algoritmasını kullanması durmunda;

100(100-1)/2=4950

adet anahtar kullanması gerekecektir. Ayrıca bir personelin şirketten ayrılması durumunda, iletişim kanalının güvenliği için tekrardan gizli anahtar seçimi yapılmalıdır. Simetrik şifreleme algoritmaları güvenliği gizli anahtar dağıtımı ile sağladıklarından Açık Anahtar Alt yapılarına uyumlu değildirler ve bu nedenle e-imza oluşturmada kullanılamazlar. Çünkü "bütünlük ve kimlik doğrulama" işlemlerini gerçekleştirememektedirler.

3.3.2 Simetrik Şifreleme Algoritmaları Avantajları

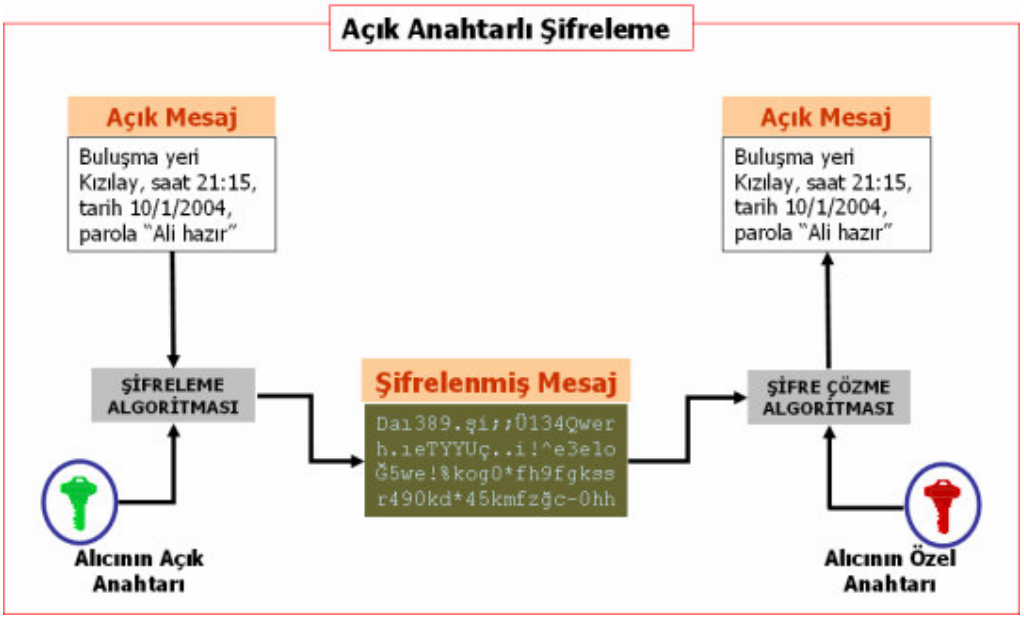
Kullandıkları matematiksel fonksiyonlar itibariyle, hızlı işleyen bir kriptolama sistemidir**.** Bilgisayar sistemleri üzerinden gerçekleştirilen şifreleme işlemlerinde simetrik şifreleme algoritmaları kullanıldığında, şifreleme ve şifre çözme işlemlerinde ihtiyaç duyulan bellek miktarı (bellek karmaşıklığı) açısından büyük avantajlara sahiptir. Sınırlı kapasiteli iletişim kanalarında kullanıldığında ileri derecede gizlilik/güvenlik sağlar.

3.3.3 Asimetrik Şifreleme Algoritmaları

İnsanoğlunun şifreleme bilimi ile ilgilendiği ilk dönemlerden 1970'li yıllarına kadar kullanılan şifreleme algoritmaları simetrik şifreleme yaklaşımlarıydı. Gizlilik esasına dayandırılan simetrik şifreleme, bu özelliğinden dolayı sadece askeri organizasyonlar ve bazı özel dar alanlarda kullanılıyordu. 1976 yılında Stanford Üniversitesinden Whitfield Diffie ve Martin Hellman adlı iki bilim insanı, geleneksel şifreleme yaklaşımlarından tamamen farklı bir şifreleme sistemi önerdiler. Diffie-Hellman şifreleme sisteminde, iletişim kanalında olan bütün şifre kullanıcılarının, biri gizli ve diğeri açık olan iki anahtarı vardır .Gizli anahtar sadece ait olduğu kişi tarafından bilinirken, açık anahtar iletişim ağındaki herkes tarafından bilinir ve açık anahtarın biliniyor olması bir güvenlik sorunu oluşturmamaktadır. İletişim ağındaki birinin genel anahtarıyla yaptığınız şifrelemeyi asla çözemezsiniz. Şifreli metni ancak açık anahtarın sahibi olan kişi, kendi özel anahtarıyla deşifre edebilir. Çünkü, gizli anahtar ile açık anahtar arasında matematiksel bir bağlantı vardır. Fakat bilinen bir anahtarla diğerini elde etmek, çok zordur. Bu kriptografik yaklaşım, "Açık-Anahtarla Şifreleme" (Publıc-Key Cryptosystem) olarak adlandırıldı. En çok bilinen asimetrik şifreleme algoritmaları; Diffie-Hellman, RSA, El Gamal, Eliptik Eğri Sistemleri gibi algoritmalardır.

Asimetrik algoritmaya dayalı bir kriptosisteminde, aynı iletişim ağı üzerindeki Alice Bob'a aşağıdaki senaryo ile şifreli bir mesaj gönderir;

- Alice ve Bob bir açık-anahtarlı şifreleme algoritması kullanma konusunda anlaşırlar (örneğin RSA).
- Bob, Alice'e açık anahtarını gönderir.
- Alice mesajını Bob'un açık anahtarıyla şifreleyip, Bob'a gönderir.
- Bob, Alice'ten gelen mesajı kendi özel anahtarıyla çözer ve orijinal mesajı elde eder.[12]



Şekil-7 Asimetrik Şifreleme (Erol,H.-Aralık 2004)

3.3.3.1 Asimetrik Şifreleme Algoritmalarının Avantajları

Asimetrik kritografi algoritmaları, şifreleme bilimi açısından, anahtar dağıtımı sorununu çözerek, şifre kullanımını günlük hayatta herkesin hizmetine sunma olanağını sağlamıştır. Asimetrik kritografinin en önemli üstünlüğü, "Açık Anahtar Alt Yapısı" üzerinde, inkar edilmezliği edilmezliği sağlayan e-imza uygulamasına uyumlu olmasıdır. Bu özellikleri ile asimetrik şifreleme yaklaşımlarının, şifreleme algoritmaları açısından devrim niteliğinde olduğu kabul ediliyor. Özellikle günlük hayattaki iş ve işlemlerin bilgisayar sistemleri ve bunlara arasındaki ağlar üzerinden çok yoğun olarak gerçekleştiği günümüzde, gerçekten de

asimetrik şifreleme algoritmasıyla geliştirilen "Açık Anahtar Alt Yapısı" ve bu yapıda inşa edilip, kullanılan e-imza mükemmel bir uygulamadır.

3.3.3.2 Asimetrik Şifreleme Algoritmalarının Dezavantajları

Asimetrik şifreleme için kullanılan matematiksel fonksiyonlar üstel fonksiyonlar olduğundan, işlem kapasiteler daha yüksektir. Dolayısıyla bilgisayar sistemlerinde, şifreleme ve şifre çözme işlemlerinde ihtiyaç duyulan bellek miktarı (bellek karmaşıklığı) açısından olumsuzluk teşkil ederler.

3.4 Elektronik Ortamlarda Güvenlik Yöntemlerinin Karşılaştırılması

Çoklu uygulamalarda açık ve neredeyse dünyanın en ücra köşesinde bile kullanılan elektronik ortamların en büyük sorunları şüphesiz ki güvenlik sorunudur. Bilgisayar sistemlerini ve ağlarını kullanan kişiler, elektronik ortamda gerçekleştirdikleri iş ve işlemlerin önem derecesine göre güvenlik politikalarını belirleyerek, bu ortamdaki güvenlik derecelerini artırmak amacını güderler. Aşağıdaki tablo elektronik ortamda kullanılan güvenlik politikalarının etkinlik alanlarıyla birlikte karşılaştırılması yapılmıştır.

	Kimlik Kanıtlama	**Gizlilik**	**Bütünlük**	**İnkar Edememezlik**
Anti-virüs			**Var**	
Güvenlik Duvarı	**Var**	**Var**		
Erişim Denetimi	**Var**	**Var**		
Şifreleme		**Var**		
Sayısal imza	**Var**		**Var**	**Var**
Açık Anahtar Alt yapı	**Var**	**Var**	**Var**	**Var**

Tablo-2 Güvenlik Yöntemlerinin Karşılaştırılması

3.5 E-imza Oluşturmada Kullanılan Şifreleme Algoritmaları

Genel olarak bir e-imza algoritmasının matematiksel ifadesi beş elemanlı bir küme üzerinden tanımlanabilir. Bu kümenin elemanları; $(\mathcal{P}, \mathcal{A}, \mathcal{K}, \mathcal{S}, \mathcal{V})$ olsun:

- $\mathcal{P}$ kümesi imzalanacak metinlerin kümesi olsun,
- $\mathcal{A}$ kümesi uygulanacak imzaların kümesi olsun,
- $\mathcal{K}$ imzalamada kullanılacak anahtarların kümesi olsun,
- Her $K \in \mathcal{K}$ için bir imzalama algoritması olan $sig_K \in \mathcal{S}$ ve imza doğrulama algoritması olan $ver_K \in \mathcal{V}$ vardır. Her sig_K: $\mathcal{P} \rightarrow \mathcal{A}$ ve ver_K: $\mathcal{P} \times \mathcal{A}$ {true,false} fonksiyonları, $x \in \mathcal{P}$ mesajlar ve $y \in \mathcal{A}$ imzalama için aşağıdaki denklem eşitliğini sağlar;

$$ver(x, y) = \begin{cases} \text{true} & \text{if } y = sig(x) \\ \text{false} & \text{if } y \neq sig(x). \end{cases}$$

Her $K \in \mathcal{K}$ için, sig_K ve ver_K fonksiyonları polinom (polynomial-time) fonksiyonlar olmalıdır. ver_K açık anahtarı (public) veren fonksiyon, sig_K imzalama fonksiyonu ise gizli (secret) anahtarı veren fonksiyon olmalı. Bu gizli anahtar imza sahibi dışında bilinmeyen bir yapı/fonksiyon özelliği taşımalıdır.[12]

Matematiksel algoritmasından da anlaşılacağı gibi, e-imzanın herhangi bir şifreleme algoritması üzerinde uygulanabilmesi için, şifreleme algoritmasının asimetrik olması gerekir. Asimetrik şifreleme algoritmaları "Açık Anahtar Alt Yapıları" itibariyle; kimlik doğrulama, gizlilik, bütünlük ve inkar edilmezlik niteliklerini sağlar. Bu özellikler, teknik açıdan yeterli ve hukuksal olarak geçerli olan e-imza özellikleridir.

3.6 RSA Şifreleme Algoritması

İlk kez 1977 yılında Ron Rivest, Adi Shamir ve Len Adleman tarfından elde edilen RSA algoritması, bu şifreleme yaklaşımını geliştirenlerin soyadlarının baş harfleriyle adlandırılmıştır. RSA asimetrik şifreleme yaklaşımı olarak, Dünya'da Açık Anahtar Alt Yapı uygulamaların en çok kullanılan algoritmadır. Diffie-Hellman adlı araştırmacılarının 1976 yılında açık anahtar alt yapılı şifreleme algoritmasını ilan etmesinde kısa bir süre sonra geliştirilen RSA, şifrelemeyi ve imzalamayı aynı anda sağlayan, güvenirlik derecesi yüksek bir asimetrik kriptografi algoritmasıdır. RSA kriptografi algoritmasının güvenliği, tamsayılarda çarpanlara ayırma probleminin kolay çözülmemesi prensibine dayanır.

3.6.1 RSA Algoritmasında Anahtar Çifti Oluşturma

- Zn = {0, 1, 2, 3, ... , n - 1} olmak üzere, asal , birbirinden farklı ve yaklaşık aynı uzunlukta ***p*** ve ***q*** sayıları seçilir,
- $n = p \cdot q$ ve $\Phi = (p - 1) \cdot (q - 1)$ değeri hesaplanır,
- $1 < e < \Phi$ ve $OBEB(\Phi, e) = 1$ olacak şekilde bir e tamsayısı seçilir,
- Öklid algoritması kullanılarak, $1<d<\Phi$ aralığında $e \cdot d \equiv 1 \pmod{\Phi}$ koşulunu sağlayacak ***d*** değeri hesaplanır,
- Özel anahtar: ***(n,d)*** ve açık anahtar: ***(n, e)***'dir.

3.6.2 RSA Algoritmasında Şifreleme ve Deşifreleme İşlemi

X şahsı ***Y***'ye ***m*** mesajını şifreleyerek göndermek istiyor. Bunun için aşağıdaki işlem adımları gerçekleştirilir.

- ***X*** önce ***Y'*** nin açık anahtarı olan ***(n, e)***'yi alır,
- Göndermek istediği mesajını **[0**, ***n-1*****]** aralığında yazar,
- Daha sonra $c \equiv m^e \pmod{n}$ işlemiyle şifreli mesajı elde eder,
- ***X*** elde ettiği ***c'***yi ***Y*** şahsına gönderir.

***Deşifreleme işlemi*:**

Y şahsı, ***X*** şahsı tarafından gönderilen ***c*** şifreli mesajını, gizli anahtarı olan ***d'***yi kullanarak;

$m \equiv c^d \pmod{n}$ işlemini uygulayıp orijinal mesajı elde eder.

3.6.3 RSA'da örnek uygulama

***Anahtar çifti oluşturma*:**

X şahsı anahtar çifti oluşturmak için:

- ***p*=**2357 ve ***q*=**2551 olmak üzere iki tane asal sayı seçer,
- $n = p \cdot q$ =2357.2551=6012707 ve $\Phi = (p - 1) \cdot (q - 1)$=(2357-1)**.**(2551-1)=6007800 değerlerini hesaplar,
- Bir tane ***e*=**3674911 değer seçer ve bu değer:$OBEB(\Phi, e) = 1$ şartını sağlar,
- Öklid Algoritmasını kullanarak; $e \cdot d \equiv 1 \pmod{\Phi}$ den: 3674911**.** $d \equiv$**1(**mod6007800) denkliğinden**,d=**422191 değerini elde eder,
- *X*'ina çık anahtarı (***n***=6012707,***e***=3674911) ve özel anahtarı (***n***=6012707,***d***=422191) olur.

***Şifreleme işlemi*:**

Y şahsı ***X'*** e ***m*=**5234673 mesajını şifreli olarak göndermek için ***X***'in açık anahtarı olan (***n***=6012707, ***e***=3674911) değerini kullanarak;

$c \equiv m^e \pmod n = 5234673^{3674911} \pmod{6012707} \equiv 3650502$ şifreli mesajını elde eder.

***Deşifreleme işlemi*:**

X gelen şifreli mesajdan orijinal mesajı elde etmek için:

$m \equiv c^d \ (mod\ n) = 3650502^{422191} \pmod{6012707} = 5234673$ işlemini yapar.[11]

3.6.4 RSA İmzalama Algoritması

RSA Algoritması hem şifreleme hem de imzalama fonksiyonları olan bir algoritma özelliği taşıdığından, şifreleme ve imzalamayı ayrı ayrı ele aldık. Aslında anahtar çiftini oluşturma algoritması aynı olmak üzere, şifreleme ve imzalama da benzer yaklaşımlarla gerçekleştirilmektedir.

İmzalama için anahtar çiftini oluşturma:

- Zn = {0, 1, 2, 3, ... , n - 1} olmak üzere, asal, birbirinden farklı ve yaklaşık aynı uzunlukta ***p*** ve ***q*** sayıları seçilir,
- $n = p \cdot q$ ve $\Phi = (p - 1) \cdot (q - 1)$ değeri hesaplanır,
- $1 < e < \Phi$ ve $OBEB(\Phi, e) = 1$ olacak şekilde bir e tamsayısı seçilir,
- Öklid algoritması kullanılarak, $1<d<\Phi$ aralığında $e \cdot d \equiv 1 \pmod{\Phi}$ koşulunu sağlayacak ***d*** değeri hesaplanır,
- Özel anahtar: ***(n,d)*** ve açık anahtar: ***(n, e)***'dir.

İmzalama işlemi:

- M mesaj uzayı ve $m \in$ M olmak üzere bir ***m*** mesajı seçilir,
- $m \in [0, n - 1]$ aralığında tamsayı olarak seçilir,
- $s \equiv m^d \pmod n$ değeri hesaplanır,
- ***s, m*** mesajının imzalanmış halidir.

***İmza doğrulama işlemi*:**

- İmzalayan kişinin açık anahtarı ***(n, e)*** öğrenilir,
- $m \equiv s^{e(} \bmod n)$ değeri hesaplanır,
- Yapılan işlem sonucunda ***m*** değeri elde ediliyorsa mesaj doğrulanmış olur.[12]

3.6.5 RSA Algoritmasının Kaba Kodu

Kabul edelim ki $n = p \cdot q$,

p ve q **asal sayı**

$P=C= Z_n$ olsun tanımlansın

$K= \{(n,p, q,a, b) : n = pq,p, q \text{ asalsayı}, a.b= 1 \pmod{\Phi(n)}\}$ verilsin

K için $K= (n, p, q, a, b)$ tanımlansın

$e_K(x) = x^b \bmod n$ ve
$d_K(y) = y^a \bmod n$
$(x, y \ \varepsilon \ Z_n)$.
Bu değerlerden n ve b açık anahtar ve p,q,a gizli anahtar olur.

3.6.6 RSA Algoritmasının Açıkları

RSA algoritması üzerine yapılan saldırılardan en önemlisi, elde edilen açık anahtar kullanılarak gizli anahtara ulaşılmaya çalışılmasıdır. Saldırgan, n katsayısının çarpanları olan p ve q değerlerini hesaplamaya çalışır. Eğer bu değerler bulunursa özel anahtara ulaşılabilir. Buradaki en zor kısım n sayısını çarpanlarına ayırma işlemidir. Ancak n değerinin yeterince büyük olmaması veya p, q çiftinin ve ayrıca e değerinin iyi seçilmemesi durumlarında RSA'nın güvenli olduğu söylenemez.

3.7 ElGamal Algoritması

ElGamal kriptografi algoritması, Diffie-Hellman algoritmasının anahtar değişimi üzerine kurulmuş bir sistemdir. ElGamal Algoritması, hem şifreleme işlemini hem de imzalama işlemini gerçekleştirebilme özelliğine sahiptir. Ayrık logaritma problemine dayandırılmış bir güvenirliğe sahiptir.

Sayılar teorisinde hesaplaması kolay fakat tersinin hesaplaması zor olan fonksiyonlar vardır. Sınırlı alanda kuvvet alma, bu fonksiyonlara verilebilecek bir örnektir. Sınırlı alan olarak sadece asal sayıları düşünelim;

p bir asal sayı ve g de Zp * de primitif bir kök olsun. O zaman kuvvet fonksiyonu

$$Exp : Z_{p-1} \longrightarrow Zp^{*}, \qquad x \longrightarrow g^{x}$$

hesaplanabilir. Kuvvet algoritması fonksiyonunun tersini hesaplamak için etkili bir algoritma bilinmemektedir. Bu tahmine ayrık logaritma problemi denir.[11],[12]

3.7.1 ElGamal Kripto Sisteminin Kaba Kodu

Kabul edelim ki p ayrık logaritma problemi için Zp de tanımlı bir asal sayı olsun,

Yine kabul edelim ki $\alpha \varepsilon Zp^{*}$ ve α primitif bir eleman olsun,

$\mathcal{P} = Zp^{*}$, $C = Zp^{*} . Zp^{*}$, olmak üzere,
$\mathcal{K}=\{(p, \alpha, a, \beta): \beta = \alpha^{a} \ modp \}$ olarak tanımlanır,

Bu değerlerden (p, α, β) açık anahtar, (a) ise gizli anahtardır,

$K= (p, \alpha, a, \beta)$ ve gizli anahtar rastsal seçilmiş (a) için, $k \varepsilon Zp-1$ de $e_K (x,k)=(y_1, y_2)$, değeri tanımlansın,

öyleki;

$y_1 = \alpha^{k} modp$ ve
$y_2 = x \beta^{k} modp$
$y_1, y_2 \varepsilon Zp^{*}$ için,

$$d_K(y_1, y_2) = y_2 (y_1^{a})^{-1} modp$$

tanımlanır.

3.7.2 ElGamal Şifreleme Algoritmasında Anahtar Oluşturma

ElGamal şifreleme algoritmasında, şifre kullanıcılarından her biri önce açık anahtarlarını ve sonra da açık anahtara bağlı olarak gizli anahtarlarını oluştururlar. Buna göre X şifre kullanıcısı aşağıdaki işlem aşamaları sonunda anahtar çiftini oluşturur:

- Yeterli büyüklükte bir ***p*** sayısını seçer,
- ***modp***'ye tamsayıların oluşturduğu çarpım grubu (***Zp **)*** nin bir jeneratörü olan ***(α)***'yı oluşturur,
- ***1≤a≤p-2*** aralığında bir **(*a*)** tam sayısını seçer ve ***(α^a modp)*** değerini hesaplar,
- ***X'*****in açık anahtarı (*p*, α , α^*a*) değeri, gizli anahtarı ise (*a*)** değeridir.

3.7.3 ElGamal Şifreleme Algoritmasında Şifreleme

Bir ***Y*** şifre kullanıcısı ***X'***e göndermek üzere şifreli bir mesaj oluşturmak isterse aşağıdaki işlem adımlarını gerçekleştirir:

- ***X'***in açık anahtarını olan (*p*, α , α^*a*) değerini alır,
- Göndereceği ***m*** mesajını {0,1,…….,***p***-1} aralığında tam sayı olarak ifade eder,
- ***1≤k≤p-2'***yi koşulunu sağlayan bir ***k*** tamsayısını seçer,
- $\gamma = \alpha^k \ modp$ **ve** $\delta = m\ (\alpha^a)^k modp$ değerlerini hesaplar,
- ***c*** $=(\gamma, \delta)$ şifreli mesajını ***X'***e gönderir.

3.7.4 ElGamal Şifreleme Algoritmasında Deşifreleme

X şahsı kendisine gönderilen şifreli mesajdan orijinal mesajı elde etmek için;

- **(*a*)** gizli anahtarı kullanılarak $\gamma^{-a} = \alpha^{-ak} modp$ değeri hesaplanır,
- $\gamma^{-a}.\delta = \alpha^{-ak} m\ \alpha^{ak} = m$ **(modp)** işlemleri uygulanır.[11],[12]

3.7.5 ElGamal Şifreleme Algoritmasında İmzalama ve İmza Doğrulama

ElGamal Algoritması ile imza oluşturma amacı, diğer algoritmalarda olduğu gibi, imza sahibinin kimliğini belirlemeye yönelik çalışır. Şifrelenmiş mesaja, kişinin imzasını da ekleyerek gönderir. ***X*** şahsının açık anahtarı ($p, \alpha, y = \alpha^a$) olmak üzere, ElGamal Algoritması ile bir imza uygulamasını gerçekleştirecekse aşağıdaki işlem sırasını takip eder:

İmzalama:

- İmzalayacağı ***m*** mesajını ***m*** ε ***Zp*** olacak şekilde seçer, eğer ***m*** mesajı ***Zp'***nin elemanı değilse, hash fonksiyonu uygulanarak, mesajın ***Zp'***nin elemanı olması sağlar,
- ***1*** $\leq$ ***t*** $\leq$ ***p-2*** ve **OBEB *(t,p-1)*=1** koşullarını sağlayan sağlayan bir ***(t)*** tamsayısı seçer,
- $r=\alpha^{t}$ *ve* $s=t^{-1}(m-ra) mod(p-1)$ işlemlerini gerçekleştirir,
- Böylece ***(m,r,s)*** ***X'*** in imzalı mesajıdır.

İmza doğrulama:

Şifreli mesajı alan ***Y*** şahsı, imzayı doğrulamak için :

- İlk olarak (r) değerinin , **1** $\leq$ ***r*** $\leq$ ***p-1*** koşulunu sağlayıp sağlamadığını kontrol eder. Eğer koşul sağlanmıyorsa imza kabul edilmez,
- $v=\alpha^{m}$ *(modp) ve* $w=y^{r}.r^{s}$ *(modp)* değerini hesaplar,
- **Eğer** $v=w$ eşitliği sağlanırsa imza doğrulanmış olur.

3.7.6 ElGamal Şifreleme Algoritmasında Anahtar Oluşturma, İmzalama ve İmza Doğrulama İçin Örnek Uygulama

Anahtar Oluşturma:

- ***X*** şahsı ***p*=2357** asal sayısı**,** α**=2** ve ayrıca ***a*=1751** değerinde bir gizli anahtar seçer.
- $(\alpha^{a} mod p)=2^{1751}$**Mod2357=1185** değerini elde eder.
- ***X'*** in açık anahtarı (p,α,α^{a}) **=(2357,2,1185)** olur.

İmzalama:

X imza kullanıcısı göndereceği mesajı ***m*=1463** olarak seçmiş olsun:

- ***X*** Önce rasgele bir ***t*=1529** gibi bir tamsayı seçer,
- $r=\alpha^{t}$ $mod p=2^{1529} \bmod 2357=1490$
- Sonra $t^{-1} \bmod(p-1)=1529^{-1} \bmod(2356)=245$ işlemi gerçekleştirir,
- Bir adım sonra da ;

- $s=t^{-1}(m-ra) mod(p-1)$ =245(1463-1490.1751)mod2356=1777 değerini elde eder,
- *X*'in imzası olan ***(m,r,s)*=**(1463**, 1490, 1777**) olur.

***İmza Doğrulama*:**

Y imza kullanıcısı ***X*** tarafından gönderilen imzalı mesajı aldığında, imza doğrulama için aşağıdaki işlem adımlarını takip eder:

- $v=\alpha^{m}\ (modp)=2^{1463} \text{mod} 2357=1072$
- $w=y^{r}.r^{s}\ (modp)=1185^{1490}1490^{1777} \text{mod} 2357=1072$
- $v=w=1072$ olduğundan imza doğrulanmıştır.

3.8 Eliptik Eğri Sistemleri

Eliptik Eğriler 1890 yıllarında matematiksel olarak keşfedilmiş olmalarına rağmen, bir şifreleme yöntemi olarak kullanılmaları 1985 yılında Neal Koblitz ve Victor Miller tarafından gerçekleştirilmiştir.

Eliptik eğri algoritması asimetrik bir şifreleme sistemi olarak, ayrık logaritma probleminin zorluğuna dayanır. Matematiksel olarak daha karmaşık bir yapıda olmasına karşın, RSA ve DSA algoritmalarının şifreleme için sağladığı güvenlik düzeyini aynı seviyede çok daha kısa parametrelerle sağlamaktadır. Şifreleme işleminde kullanılan parametre değerleri küçüldükçe yapılan işlemlerin süresi de azalmaktadır. Bu durum, imzalama ve imza doğrulama işlemlerinin daha hızlı bir şekilde gerçekleştirilmesi anlamına gelir.

RSA ve Eliptik Eğri algoritması ile yapılan şifrelemenin bit düzeyinde karşılaştırılması aşağıdaki gibidir.[13],[17]

bit olarak istenilen güvenlik düzeyi	**Eliptik Eğri Şifreleme yönteminde gerekli anahtar uzunluğu**	**RSA şifreleme yönteminde gerekli anahtar uzunluğu**
56	112	512
80	160	1024
112	224	2048
128	256	3072
192	384	7680
256	512	15360

Tablo-3 Aynı Güvenlik Seviyesinde Karşılaştırmalı Anahtar Boyları (Menzes,1993),[15]

"Eliptik eğri DSA (EDSA – Elliptic Curve DSA), DSA'nın eliptik eğri kullanılarak meydana getirilmiş bir benzeridir. 1992 yılında hazırlanmış ve 1999 yılında ANSI (American National Standards Institute – Amerika Ulusal Standartlar Enstitüsü), daha sonra ise IEEE (Institute of Electrical and Electronics Engineers – Elektrik ve Elektronik Mühendisleri Enstitüsü) ve ISO (International Organization for Standardization – Uluslararası Standardizasyon Teşkilatı) tarafından standart olarak kabul edilmiştir".[14]

3.8.1 Matematiksel Olarak Eliptik Eğri Tanımı

F karekteristiği 2 veya 3 olmayan bir cisim olsun. F üzerindeki E ile gösterilen bir eliptik eğri:

$$y^2 = x^3 + ax + b \qquad (1)$$

şeklinde olup bu denklemde $a, b \in F$ dir ve $4a^3 + 27b^2 \neq 0$ olmalıdır .Eğer K, F' yi kapsayan başka bir cisim ise E' nin K noktaları, $E(K)$ ile gösterilir. (1) denkleminin bütün $(x, y) \in K \times K$ şeklindeki çözümleri ile birlikte gösterimi ∞ olan özel sonsuzdaki noktadır. $E(K)$ kümesi ∞ ile birlikte toplamsal olarak yazılan bir değişmeli grup oluşturur.

Eğer $P = (x_1, y_1) \in E$ ise $-P = (x_1, -y_1)$ dir.

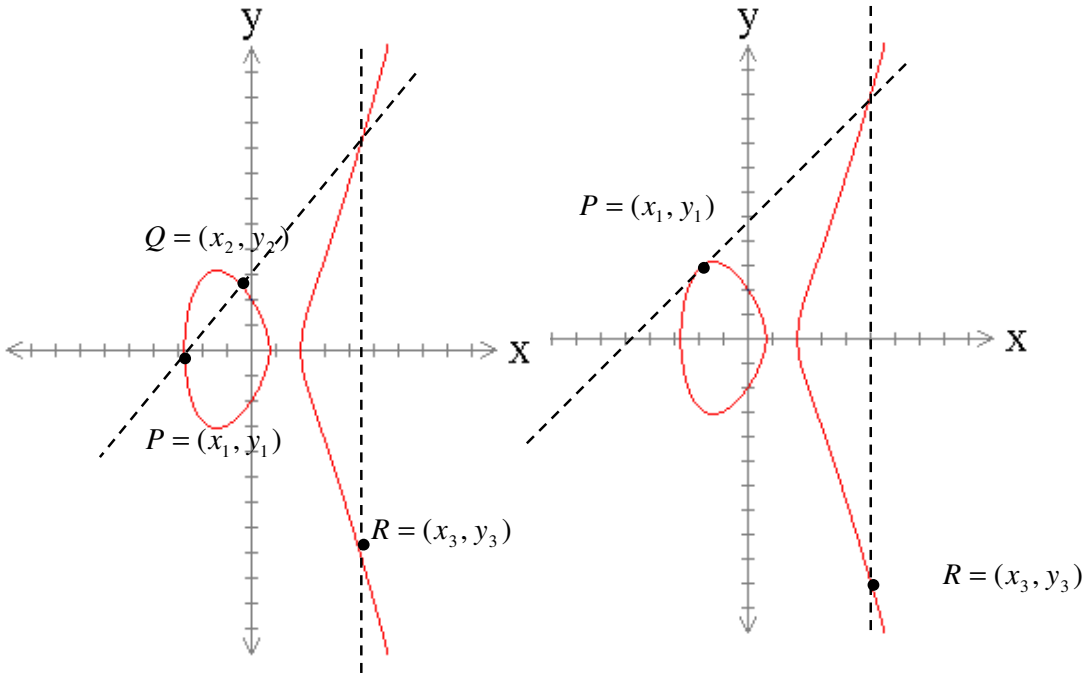
Eğer $Q = (x_2, y_2) \in E$, $Q \neq -P$ ise $P + Q = (x_3, y_3)$ olup burada

$$x_3 = \lambda^2 - x_1 - x_2,$$

$$y_3 = \lambda(x_1 - x_3) - y_1,$$

ve

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1}, P \neq Q \\ \dfrac{3x^2 + a}{2y_1}, P = Q \end{cases}$$



(a) P + Q = R (b) P + P = R

Sonlu cisimler üzerinde tanımlanan eliptik eğriler iki gruba ayrılırlar. Bunlar, süpersingüler (supersingular) eğriler ve süpersingüler olmayan (non-supersingular) eğrilerdir. Bu eğrilerin büyük çoğunluğunu süpersingüler olmayan eğriler oluşturur. Eğer F karakteristiği 2 olan sonlu bir cisim ise süpersingüler eliptik eğri, E,

$$y^2 + cy = x^3 + ax + b, \quad a, b, c \in F \ \ ve \ \ c \neq 0,$$

şeklindedir. Süpersingüler olmayan eğriler ise

$$y^2 + cy = x^3 + ax^2 + b, \quad a, b \in F \ \ ve \ \ b \neq 0,$$

biçimindedir. İki durumda, $K \supset F$ olmak üzere $E(K)$ noktalar kümesi ∞ ile birlikte değişmeli grup oluştururlar. ∞ noktası grubun birim elemanıdır.[16]

3.8.2 Eliptik Eğri Kriprosistemleri

Eliptik Eğri Kriprosistemi daha önce de belirtildiği gibi "Eliptik Eğri Ayrık Logaritma Problemi (EEALP)" problemine dayanır. Bu problemin tanımı şöyle yapılır:

- F_q üzerinde bir $\boldsymbol{E}$ eliptik eğrisi verilsin,
- Mertebesi N ve $P \in E(F_q)$ olan bir nokta belirlensin,

- $Q \in E(F_q)$ olan başka bir nokta daha verilsin,

$Q = xP$ eşitliğini sağlayan $0 \le x < N$ aralığındaki x tamsayısının bulma problemine "***Eliptik Eğri Ayrık Logaritma Problemi (EEALP)"*** denir.[17]

3.8.3 Diffie-Helman Algoritmasıyla Eliptik Eğri Üzerinde Anahtar Değişimi

Bilindiği gibi Diffie-Helman Algoritması açık anahtarlı kriptosistemlerinde sadece anahtar değişimini gerçekleştirmekte olup, şifreleme yeteneği bulunmamaktadır. Bu algoritmanın sağlamış olduğu anahtar değişimi olayı eliptik eğriler üzerinde de gerçekleştirilebilir.

Eliptik Eğri Diffie-Helman Algoritmasıyla Anahtar Değişimi şöyle açıklanabilir:

- E , F_q da bir eliptik eğri olsun,
- P herkesçe bilinen eğri üzerinde bir nokta olsun,
- A kişisi gizli tutmak üzere bir rassal k_A tamsayısı seçer ve $k_A P$ noktasını hesaplayıp B kişisine gönderir,
- Benzer şekilde B kişisi gizli bir rassal k_B tamsayısı seçip $k_B P$ sayısını A kişisine gönderir,
- Bu iki kişinin mesajlaşmada kullanacakları ortak anahtarları $Q = k_A k_B P$ dir.[16]

3.8.4 ElGamal Algoritmasıyla Eliptik Eğri Üzerinde Şifreleme

ElGamal Algoritmasıyla Eliptik Eğri üzerinde şifreleme işlemini gerçekleştirmek isteyen bir ***A*** kişisi, bir ***B*** kişisine bir mesaj göndermek isterse aşağıdaki işlem adımlarını takip eder:

- Mesaj kümesinin her bir elemanı kararlaştırılmış bir metodla E'ye indirgenir ve Artık ***m*** mesajının E'nin bir elemanı olması sağlanır,
- ***A*** kişisi EEDHP'de olduğu gibi gizli bir rassal k_A pozitif tamsayısı seçip $k_A P$ sayısını açık anahtarı olarak ilan eder,
- A'ya mesaj göndermek isteyen ***B*** kişisi gizli bir rassal ***l*** pozitif tamsayısı seçer ve ***A*** kişisine $(lP, M + l(k_A P))$ ikilisini gönderir,

- ***A*** kişisi mesajı deşifre etmek için ikilinin birinci bileşenini alır ve kendisinin bildiği gizli k_A sayısı ile çarpar. Buradaki şifreleme sisteminin güvenirliği eliptik eğrilerde Diffie-Halman Algoritmasının anahtar değişimi problemine dayanır. [16]

3.8.5 Eliptik Eğri Şifreleme Algoritmasıyla E-imza Oluşturma

Eliptik Eğri Algoritmasıyla e-imza oluşturmak isteyen bütün kullanıcılar aşağıdaki işlem adımlarını takip ederler:

- ***P*** noktası mertebesi asal ***N*** olan bir nokta olsun,
- Her kullanıcı [1, N - 1] aralığında rassal bir x tamsayısı seçer,
- $Q=xP$ noktası bu kullanıcının açık anahtarı, x ise gizli anahtarıdır,
- Özet (*hash)* değeri **H**, $1 < H < N - 1$ olarak elde edilir,
- [1, N - 1] aralığında rassal bir ***k*** tamsayı seçilir,
- $kP = (x_1, y_1)$ noktası hesaplanır ve $r = x_1 \bmod N$ değeri atanır. Eğer $r = 0$ çıkarsa birinci adıma dönülür.
- $k^{-1} \bmod N$ hesaplanıp $s = k^{-1}(H + xr) \bmod N$ değeri bulunur. Eğer $s = 0$ olursa birinci adıma tekrar gidilir.

Mesajın eliptik eğri kullanılarak oluşturulan imzası (r, s) ikilisidir. [14],[17]

3.8.6 Eliptik Eğri Şifreleme Algoritmasıyla E-imza Doğrulama

- Mesajı imzalayıp gönderen kişinin Q açık anahtarı alınır,
- r ve s sayılarının [1, N - 1] aralığında olduğu doğrulanıp, mesajın H özet (hash) değeri hesaplanır,
- $u_1 = s^{-1}H \bmod N$ ve $u_2 = s^{-1}r \bmod N$ hesaplanır.
- $u_1P + u_2Q = (x_0, y_0)$ hesaplanır ve $v = x_0 \bmod N$ bulunur.
- Eğer v = r ise imza doğrulanır. [13],[17]

3.8.7 Şifreleme Güvenliğinde Eğrilerin Seçimi

Eliptik eğri imzalama algoritmaları, üzerine inşa edildikleri şifreleme algoritmalarının (RSA, DSA, ElGamal gibi) güvenlik özelliklerini taşımaktadır. Güvenlik kriterleri seçilen eliptik eğrilerin ve kullanılan parametrelerin iyi bir şekilde seçilmesini gerekli kılıyor.[13],[14]

Şifrelemenin güvenlik seviyesini arttırmak için kullanılacak olan eliptik eğrileri belirlerken, seçilen eğrilerin bilinen tüm ataklara karşı dirençli olmasına dikkat edilmelidir.

- Pollard-ρ atağına karşı koymak için $\#E(F_q)$'nun yeterince bütük bir N asal çarpanı olmalıdır. (Örn.: $N > 2^{160}$)
- Weil ve Tate ikili atakları için N'nin $q^k - 1$ sayısını $1 \le k \le C$ için bölmemesi gerekir. Burada $C = 20$ yeterlidir.
- Eğer q asal ise $\#E(F_q)$ değeri q'ya eşit olmalıdır.[16],[17]

3.9 Diffie-Hellman Anahtar Değişimi

Kriptografi tarihinde simetrik şifreleme yaklaşımındaki en büyük sorunlardan biri olan anahtar dağıtımı sorununa çok iyi bir düzeyde çözüm getiren, şifreleme biliminin, büyük oranda toplumsal alandaki kullanımını sağlayabilen, ilk açık anahtarlı algoritmadır. Daha önce de belirttiğimiz gibi şifreleme bilimi açısından tam bir devrim olma özelliğine sahiptir. Diffie ve Hellman`ın 1976 yılında "New Directions in Cryptography" isimli makalelerinde yer alan bu şifreleme yaklaşımı, "Açık anahtar altyapılı şifreleme sistemi" olarak adlandırıldı. Diffie-Hellman Anahtar değişimi özellikle ticari uygulamalarda çok rağbet gördü. Bu algoritmanın temel hedefi, iki kullanıcının bir anahtarı güvenli şekilde birbirlerine iletmeleri ve daha sonrasında da bu anahtar yardımı ile şifreli mesajları birbirlerine gönderebilmelerini sağlamaktı. Özellikle belirtelim ki, algoritma yalnız anahtar değişimini yapabilmektedir. Diffie-Hellman Algoritmasının şifreleme yeteneği yoktur. Diffie-Hellman ortak gizli anahtar oluşturma sistemi ayrık logaritma problemini üzerine kurulmuş ve güvenirliği çok büyük asal sayıları seçmeye dayanmaktadır.

Diffie-Hellman algoritmasıyla açık anahtar değişimini gerçekleştirmek isteyen ***X*** ve ***Y*** kişileri aşağıdaki işlem adımlarını takip ederler:

- ***X*, $0 \le a \le p-2$** eşitsizliğini sağlayan ve rastsal olan bir a sayısını seçer.
- **$c = g^a \pmod p$**'yı hesaplar ve bunu ***Y***'ye gönderir.
- ***Y*, $0 \le b \le p-2$** eşitsizliğini sağlayan ve tesadüfi olan bir b sayısı seçer.

- **$d = g^b \pmod p$**'yı hesaplar ve bunu ***X*** 'e gönderir.
- ***X***, ortak anahtar k' yı şu şekilde hesaplar:

$$k = d^a = (g^b)^a$$

- ***Y***, ortak anahtar k' yı şu şekilde hesaplar:

$$k = c^b = (g^a)^b$$

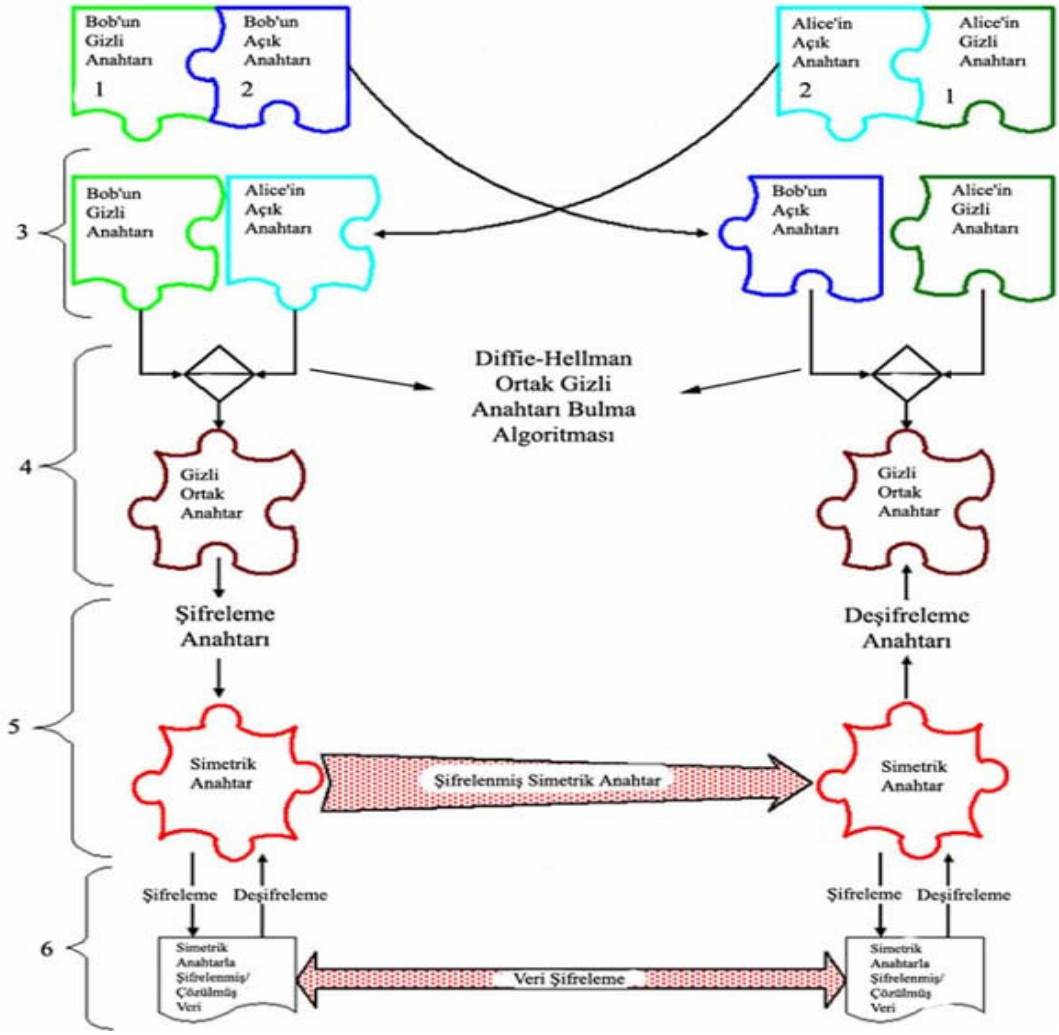
- $k = (g^b)^a = (g^a)^b$ ortak açık anahtar değeri elde edilmiş olur.

3.9.1 . Diffie-Hellman Algoritması İçin Örnek Uygulama

X kişisi ile ***Y*** Diffie-Hellman Algoritmasına göre bir ortak anahtar oluşturmak istesinler:

- Öncelikle ortak anahtar oluşturmak için ***p*=541 ve *g*=2** sayıları ortak seçilmiş olsun,
- ***X*** kişisi kendi gizli anahtarı olan**, *a* =137** sayısını seçer,
- ***Y*** kişisi kendi gizli anahtarı olan **, *b* = 193** sayısını seçer,
- ***X*,** $c = g^a \pmod p = 2^{137} \bmod 541 = 208$ değerini elde eder**,**
- ***Y*,** $d = g^b \pmod p = 2^{193} \bmod 541 = 195$ değeri eder,
- **c** ve **d** değerleri hesaplandıktan sonra **X** ve **Y** tarafından birbirilerine gönderilir. Ortak anahtar aşağıdaki işlemlerle bulunur:

$$k = c^b = (g^a)^b (\text{mod} p) = (2^{137})^{193} = \text{mod} 541) = (208)^{193} (\text{mod} 541) = 486 (\text{mod} 541)$$



Şekil-8 Diffie-Hellman algoritmasında anahtar değişimi (Yerlikaya,T. ,Buluş,E. ,Buluş,N)

3.10 DSA Algoritması

"DSA, 1991 yılında NIST tarafından Elektronik İmza Standardı (DSS – Digital Signature Standard) olarak federal uygulamalarda kullanılmak üzere oluşturulmuştur. DSA, ElGamal algoritmasının farklı bir versiyonu olup ayrık (kesikli) logaritma problemini temel almaktadır. İmzalama işleminde, özetleme algoritması olarak SHA-1 kullanılması gerekmektedir."[21] DSA ilk versiyonlarında anahtar uzunluğu en fazla 512 bit uzunluğu ile sınırlandırılmıştır. Ancak 2001 yılında DSA algoritmasında güvenlik seviyesini arttırmak için, anahtar uzunluğunun değeri 1024 bit seviyesine çıkarılmıştır.

DSA Algoritmasıyla anahtar çifti üretme, imzalama ve imza doğrulama işlemleri aşağıda verilmiştir:

Anahtar çifti üretme:

- $2^{1023} < p < 2^{1024}$ olacak şekilde bir ***p*** asal sayısı seçilir,
- (p - 1)'in asal böleni olmak üzere, $2^{159} < q < 2^{160}$ aralığında bir ***q*** asal sayısı seçilir,
- $1 < h < p - 1$ aralığında ve $h^{(p-1)/q} \bmod p > 1$ şartını sağlayan bir h değeri seçilir,
- $g = h^{(p-1)/q} \bmod p$ değeri hesaplanır,
- $0 < x < q$ aralığında rastgele bir x tamsayısı seçilir,
- $y = g^x \bmod p$ değeri ayrıca hesaplanır,
- Açık anahtar **(p, q, g, y),** özel anahtar **(x)**'dir.

İmzalama:

- M mesaj uzayı ve $m \in M$ olmak üzere bir m mesajı seçilir,
- m' = SHA-1 (m) hesaplanır,
- $0 < k < q$ olmak üzere k tamsayısı seçilir.
- $r = (g^k \bmod p) \bmod q$ hesaplanır,
- $s = (k^{-1}(m' + x \cdot r)) \bmod q$ hesaplanır,
- **(r, s)**, m mesajının imzalanmış halidir. Burada (r ve s) sıfırdan farklı değerler almalı.

İmza doğrulama:

- m mesajını imzalayan kişinin açık anahtarı **(p, q, g, y)** edinilir,
- r ve s değerlerinin $(0 < r < q)$ ve $(0 < s < q)$ aralıklarında olup olmadığı sorgulanır. (r ve s değerleri bu aralıklarda değilse imzalı mesaj red edilir, doğruysa, imza doğrulama işlemine devam edilir.)
- m mesajının bütünlük sorgusu için, m' = SHA-1 (m) hesaplanır,
- $w = s^{-1} \bmod q$ hesaplanır,
- $u = (w \cdot m') \bmod q$ ve $v = (r \cdot w) \bmod q$ hesaplanır,
- $v = ((g^u y^v) \bmod p) \bmod q$ değeri hesaplanır,

- v = r ise imza doğrulanmış olur.[21]

DSA Algoritması imza oluşturma performansı RSA Algoritmasında daha iyi iken, imza doğrulama işlemindeki performansı RSA Algoritmasına göre daha düşüktür.

DÖRDÜNCÜ BÖLÜM
E-İMZADA KULLANILAN HASH (ÖZETLEME) ALGORİTMALARI

4.Özetleme Fonksiyonları

Elektronik imza, genel ve özel anahtarların aynı kişiye ait olduğu asimetrik şifreleme algoritmalarıyla gerçekleştirilir. E-imza kullanıcısının genel anahtarı ya da başka bir ifadeyle açık anahtarı herkesçe bilinir ve bu durum imza sahibi açısından önemli bir sakınca doğurmaz. Çünkü açık anahtardan yola çıkarak, gizli anahtarı elde etme işlemi matematiksel olarak çözülmesi son derece zor bir problemdir. Asimetrik kriptosistemleri e-imza oluşturma ve doğrulama yeteneklerine sahip olma avantajlarına rağmen, simetrik şifreleme sistemlerine göre çok yavaş işleyen algoritmalardır. Örneğin; RSA, DSA, Eliptik eğri imza algoritmaları matematiksel fonksiyon nitelikleri itibariyle "Ayrık Logaritma Problemine" dayanmakta olup ($h=g^k$ eşitliğinde g ve h biliniyorken; k tamsayısını bulmaya çalışmak gibi), logaritmik eşitlikte üstel değeri bulmak gerçekten çok zor bir problemdir. Bilgisayar işlemcisi ve belleği açısından üstel fonksiyonlar daha fazla (dolayısıyla daha uzun zaman) işlem hacmine sahipler. Dolayısıyla gerçekleştirilen işlem daha uzun zamanda yapılmış olur. [19]

Buna rağmen gerek sunduğu kriptoanaliz direnci, gerekse de anahtar dağıtım kolaylıkları açısından açık anahtar tabanlı algoritmalar tercih edilmektedir.

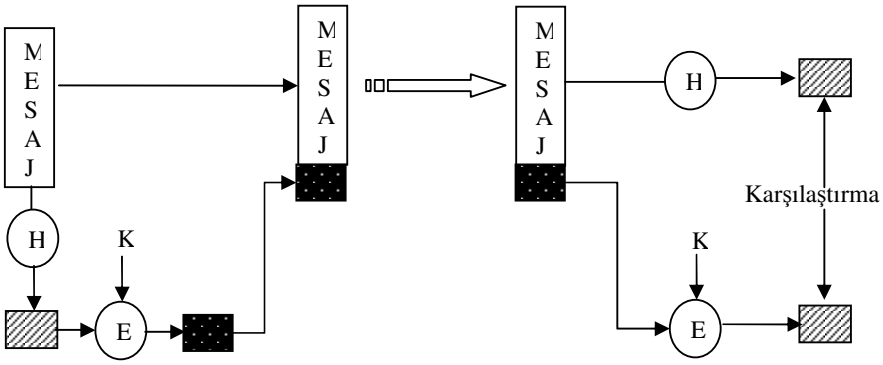
"İşlem süresini azaltmak için standartlaşmış hızlandırıcı mekanizmalar kullanılmaktadır. Sayısal imzalamada metnin kendisi değil de tek blok halinde özü (hash) imzalanmaktadır. Şifrelemede metin daha hızlı olan bir simetrik şifreleme algoritması ile şifrelenmekte, bu şifrelemede kullanılan anahtar ise açık anahtar tabanlı bir algoritma ile şifrelenmektedir. Böylelikle hem açık anahtar tabanlı

sistemlerin hız sorunu kısmen de olsa aşılmış olmakta, hem de anahtar dağıtım sorunu çözülmüştür."[20]

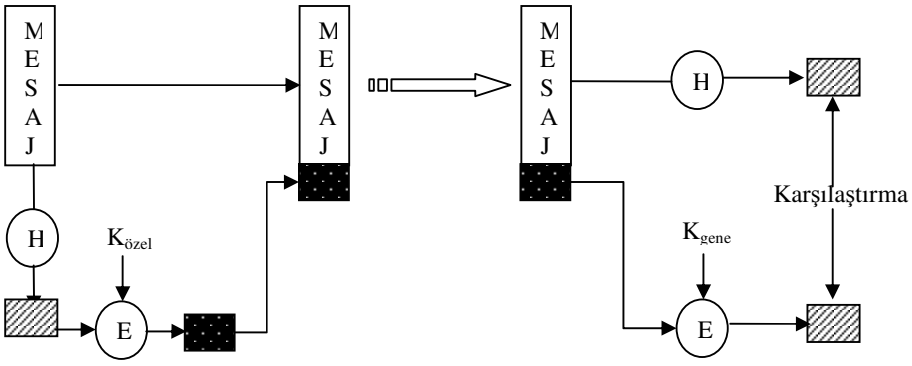
4.1 Tek Yönlü Hash Fonksiyonu

Hash Fonksiyonu giriş olarak uzunluğu değişebilen bir mesaj alıp, H(M) olarak göstereceğimiz sabit uzunluklu bir mesaj özetini oluşturur. Bir mesajın doğrulanması için, mesaj özeti asıl olmak üzere mesajla beraber mesaj özeti de gönderilir.

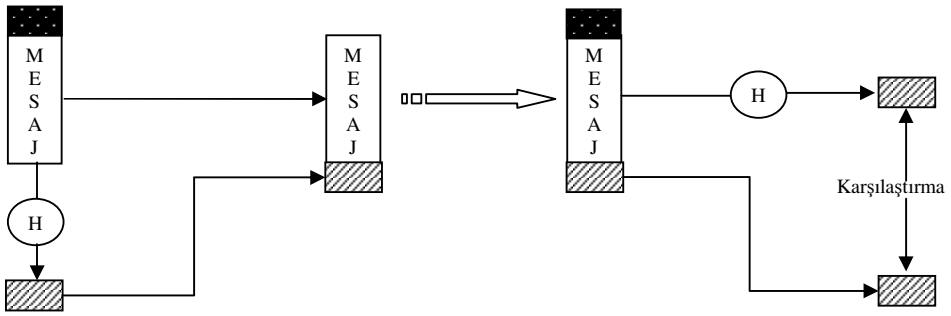
Eğer şifreleme anahtarı sadece alıcı ve gönderici tarafından biliniyorsa mesaj özeti geleneksel şifreleme ile oluşturulabilir. Mesaj ayrıca genel anahtar (public-key) kullanılarak ta şifrelenebilir. Public-key kullanmanın iki avantajı vardır; mesaj doğrulamak için e-imzanın kullanılmasına olanak sağlaması ve iletişim halindeki taraflar arasında anahtarların dağıtılması problemini ortadan kaldırmasıdır.



(a) Geleneksel şifreleme



(b) Genel anahtarlı şifreleme



(c) Özel değerle şifreleme

Şekil -9 Hash fonksiyonlarının Mesaj Özeti oluşturması (Stallings,W 1999)

4.2 Güvenli Hash Fonksiyonları ve HMAC

Tek yönlü hash fonksiyonu veya güvenli hash fonksiyonu sadece mesaj doğrulamada değil aynı zamanda e-imzada da önemli bir yere sahiptir. Bu paralelde en önemli hash fonksiyonu SHA-1 dir.

4.3 Hash Fonksiyonunda Olması Gereken Özellikler

Hash fonksiyonunun amacı, bir mesajın, dosyanın veya diğer tür bir verinin parmak izini oluşturmaktır. Bir H hash fonksiyonunun bir mesajın doğrulamasında başarılı olabilmesi için aşağıdaki özelliklere sahip olmalıdır:

- H herhangi bir uzunluktaki mesaja uygulanabilmelidir.
- H sabit uzunluklu bir çıkış üretmelidir.
- H(x) herhangi bir x bloğu için kolayca hesaplanamamalıdır.
- Verilen herhangi bir h kodu için, H(x) = h eşitliğinin x değişkeni elde edilebilirliği olanaksız olmalıdır.
- Verilen herhangi bir veri bloğu, x, için x'ten farklı H(x)=H(y) özelliğini sağlayan y kolaylıkla bulunamamalıdır.
- H(x) = H(y) iken (x, y) çiftini bulmak olanaksız olmalıdır.

İlk üç gereksinim bir hash fonksiyonun mesaj doğrulamak gibi pratik uygulamaları içindir. Dördüncü özellik, verilen bir mesaj için kolayca kod üretebilmek içindir. Bu özellik Şekil 9-c de gösterildiği gibi mesaj doğrulama

tekniğinde bir gizli katkı değer kullanıldığında önem kazanır. Beşinci özellik ile de , verilen bir mesajın hash değeri ile alternatif bir mesaj oluşturmanın mümkün olmamasını sağlar. Eğer bu özellik olmasaydı bir saldırgan şu adımları izleyerek alternatif mesaj üretebilirdi; önce mesajı hash değeri ile gözlemleyip, sonra mesajdan şifrelenmemiş hash kodunu üretecek, en sonunda bu hash kodu ile alternatif mesajı üretecektir.

Yukarıdaki listede sıralanan ilk beş özellik daha çok zayıf bir hash fonksiyonun özellikleridir. Altıncı özellikle birlikte zayıf hash fonksiyon güçlü bir hash fonksiyonuna dönüştürebilir. Altıncı özellik akıl atak sınıfından Birthday atağına karşı koruma sağlar.

Bu özelliklerin hiçbiri keskin tanımlar değildir. Hash fonksiyonu en genel isimdir ve kullanıldığı yere, duyulan ihtiyaca göre isimlendirme yapılmaktadır.

Bunun yanında bu özelliklere alternatif isimler de verilebiliyor. Bazı kaynaklarda 5. özellik için zayıf çakışma direnci (weak collision resistance), 6. özellik için güçlü çakışma direnci (strong collision resistance) denilmektedir.

Tek-yönlü hash fonksiyonu (one-way hash function); bir hash fonksiyonunun ([1],[2] ve [3]) kendi özelliklerinin yanında [4] ve [5] numaralı özellikleri göstermesidir. Çakışma-dirençli hash fonksiyonu (collision resistant hash function) ise, bir hash fonksiyonunun ([1],[2] ve [3]) özelliklerinin yanında [5] ve [6] numaralı özelliklerini de göstermesidir. Bu tip özetleme fonksiyonlarında kendiliğinden [5] numaralı özelliği göstermesi beklenir, fakat bu zorunlu değildir . Bunun yanında pratikteki neredeyse bütün uygulamalar için bu özelliğin varlığı da aranmaktadır. Onun için genel olarak hash fonksiyonu dediğimizde bütün bu özelliklerin kastedildiği düşünülmektedir. [12],[18]

4.3.1 Hash Fonksiyonlarının Genel Yapıları

Değişken uzunlukta girdileri kabul edip sabit uzunlukta bir çıktı verme özellikleri, genel olarak hash fonksiyonlarının, veri sıkıştırma fonksiyonlarının mantığından yararlanılmıştır.

Değişken m-bit uzunluğunda bir M verisinden, fonksiyon çıktısı olarak sabit n-bit elde etmek için k adet m-bit küçük veri bloklarına (m1,m2,...,mk) bölünerek işleme sokulur. Her işlemin çıktısını

$$hi = f(mi , hi-1)$$

fonksiyonu ile tanımlayalım ve (i) işlem (basamak) numarası olsun. Yani daha sade bir dille anlatacak olursak: her işlemin sonucu, bir önceki işlemin sonucu ile beraber o basamaktaki veri bloğunun tekrar işleme sokulması ile elde edilsin. O zaman bütün M verisinin özetinin oluşturulması son blok da bu işleme sokulduğunda elde edilen sonuç olacaktır. [18]

4.3.2 Mesaj Belgeleyici Kodlar (MAC - Message Authentication Codes)

Hash fonksiyonlarının bir gizli anahtarla birleştirerek kullanımı, bir verinin veya mesajın, içeriğinin o anahtarı paylaşanlardan biri tarafından gönderildiği gibi olduğunun belgelenmesi (mesaj içeriğinin değiştirilmediğinin görülmesi) mesaj bütünlüğün sağlandığını deklare eder. Hash fonksiyonlarının veri bütünlüğünün (integrity) kontrolü için kullanılmaları durmunda sadece anahtar bilgisine sahip tarafların mesajın bütünlüğünü ve kaynağını kontrol edebilmesine müsade etmektedir.

Karşılıklı iki tarafın mesajın kaynak ve bütünlüğünü doğrulaması dışında MAC'ler kişinin bir veriyi saklarken bilinçli olarak değiştirilmediğini de doğrulayabilmektedir. Normalde bir hash fonksiyonu kazara olan değişiklikleri kontrol edebilirken, kasıtlı yapılmış bir değişikliği ilgili hash çıktısı da rahatlıkla saldıran tarafından tekrar üretilebileceği için ortaya çıkaramamaktadır. Bu bir virus ya da kişisel saldırı olabilir. Fakat saldırıyı gerçekleştiren kişi ya da program (virus) gizli anahtarı bilmediği için ilgili MAC'i uygun şekilde değiştiremeyecektir.

Bunun için çok çeşitli yöntemler bulunmaktadır. Fikir vermesi için çok temel bir yöntemi örnek olarak görelim: Taraflar K gizli anahtarını paylaşıyor olsunlar. Karşı tarafa "m" mesajı gönderilirken yanında MAC fonksiyonumuzun çıktısı M(K,m) de eklenir.

Özel bir MAC fonksiyonu kullanmak dışında, bir hash fonkisyonunu MAC olarak kullanabilmek için basit olarak m mesajının sonuna K anahtarını ekleyebiliriz. O zaman M(K,m)=H(m|K) olacaktır. Fakat bu şekildeki bir kullanım bir çok güvenlik açığını da beraberinde getirmektedir. Alınabilecek önlemler arasında hash değerini H(K,m,K) olarak hesaplamak yada iki farklı anahtar kullanmak, H(K1,H(K2,M)), olarak önerilebilir.[18],[12]

4.3.3 Basit Hash Fonksiyonları

Bütün hash fonksiyonları aşağıdaki genel prensipleri kullanarak çalışır. Giriş (mesaj, dosya) bir dizi n-bit blok halinde kullanılır. Giriş değeri olarak bloklar alınarak iterative fonksiyonlarla n-bit özet elde edilir.

En basit hash fonksiyonu, her bloğun bit-bit XOR sonucunun elde edilmesidir. Bu hash fonksiyonu aşağıdaki gibi ifade edilebilir:

$$C_i = b_{i1} \oplus b_{i2} \oplus \cdots \oplus b_{im}$$

Öyle;

C_i = hash fonksiyonun i. nci biti , $1 \le i \le n$

m = girişteki n-bit blok sayısı

b_{ij} = j. ninci blokun i. ninci biti

$\oplus$ = XOR işlemi.

Tablo-4 te yukarıda ifade edilen işlem gösterilmiştir. Burada her bit, bir bloğun elemanı olarak ele alınmıştır.[18],[22]

	1. bit	2. bit	•••	n. bit
1. Blok	b_{11}	b_{21}		b_{n1}
2. Blok	b_{12}	b_{22}		b_{n2}
	▪	▪	▪	▪
	▪	▪	▪	▪
	▪	▪	▪	▪
m. Blok	b_{1m}	b_{2m}		b_{nm}
Hash Kodu	C_1	C_2		C_n

Tablo-4 XOR ile Basit Hash fonksiyonunun Kullanımı (**Stallings,W 1999**)

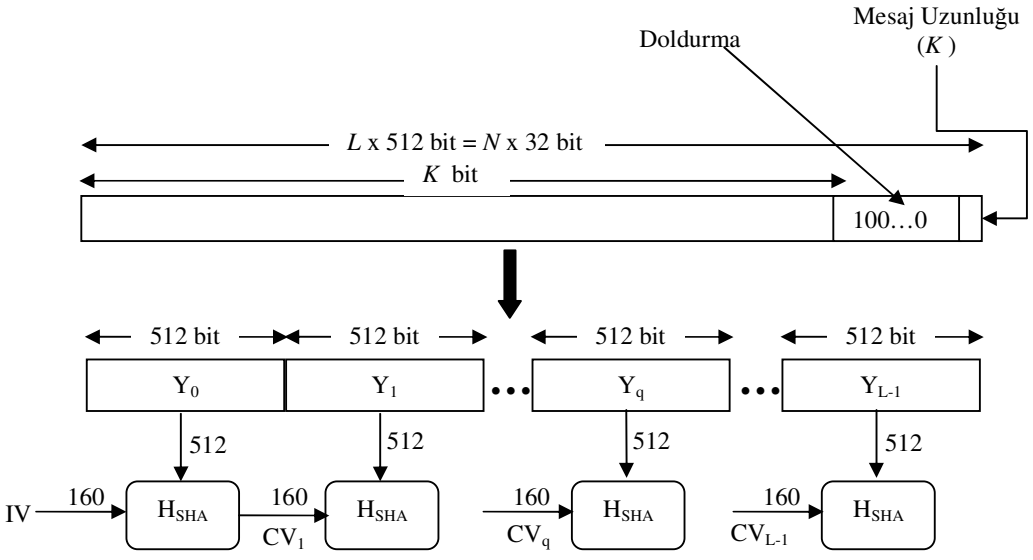
4.4 SHA-1 Özetleme (Hash) Fonksiyonu

SHA-1 fonksiyonu 2^{64} ten küçük uzunlukta bir mesajı alıp, 160 bit uzunluğunda mesaj özeti oluşturur. Damgard/Merkle iteratif algoritmasına göre mesaj 512 bitlik veri bloklarına çevrildikten sonra SHA-1 fonksiyonuna giriş olarak alınır. Algoritmadaki her iterasyonda, fonksiyon giriş olarak bir zincirleme değer ve mesajın 512 bitlik bloklarını 32 bitlik 16 oktatlar şeklinde alıp, çıkış olarak 160 bitlik yeni bir zincirleme değerini üretir. İlk zincirleme değeri (IV) her biri 32 bitlik 5 değerlerden oluşur. Son zincirleme değeri de hash fonksiyonunun ürettiği özet sonucu olan 160 bitlik değerdir. SHA-1 fonksiyonun matematiksel ifadesi aşağıda gösterilmiştir:

$$X_1 = IV \oplus D_K(Y_1)$$
$$X_i = Y_{i-1} \oplus D_K(Y_i)$$
$$X_{N+1} = Y_N \oplus D_K(Y_{N+1})$$

X_{N+1} ifadesi SHA-1 fonksiyonun sonuç hash kodu olup, ara 160 bitlik hash kodlarının toplamına eşit olduğu aşağıdaki gibi ifade edilir:

$$X_{N+1} = X_1 \oplus X_2 \oplus \cdots \oplus X_N$$
$$= (IV \oplus D_K(Y_1)) \oplus (Y_1 \oplus D_K(Y_2)) \oplus \cdots \oplus (Y_{N-1} \oplus D_K(Y_N)) \quad [18]$$



Şekil-10 SHA-1 kullanılarak mesaj özetinin oluşturulması (Stallings,W 1999)

4.4.1 SHA-1 Fonksiyonunun Kaba Kodu

Adım 1:

/* ilk 32 bitlik 16 oktatımızı oluşturalım, $m_0, m_1, \cdots m_{15}$

/* $M[t]$ mesajımızın t. ninci 8 bitlik word olsun

$t = 0; t < 16; t++$
{
$m_t = M[t*4] << 24$
$m_t = m_t \vee (M[t*4+1] << 16)$
$m_t = m_t \vee (M[t*4+2] << 8)$
$m_t = m_t \vee M[t*4+3]$
}

Adım 2:

/* şimdi diğer 32 bitlik oktatlarımızı oluşturalım, $m_{16}, m_{17}, \cdots m_{79}$

$t = 16; t < 80; t++$
$m_t = (m_{i-3} \oplus m_{i-8} \oplus m_{i-14} \oplus m_{i-16}) << 1$

/* 5 adet geçici 32 bitlik değişken tanımlayıp ilk değerlerini atayalım

$a_0 = h_1 = 0x67452301$
$b_0 = h_2 = 0xefcdab89$
$c_0 = h_3 = 0x98badcfe$
$d_0 = h_4 = 0x10325476$
$e_0 = h_5 = 0xc3d2e1f0$

Adım 3:

/*Şimdi 20'şer adımlarla zincirleme değerleri(yeni *a, b, c, d, e*) ve ara hash kodlarımızı üretebiliriz.

$t = 1; t <= 80; t++$

$a_t = (a_{t-1} << 5) + f_t(b_{t-1}, c_{t-1}, d_{t-1}) + e_{t-1} + m_{t-1} + k_t$

$b_t = a_{t-1}$

$c_t = b_{t-1} << 30$

$d_t = c_{t-1}$

$e_t = d_{t-1}$

Adım 4:

/*Sonuç olarak artık 160 bitlik hash kodu üretilebiliriz.

$h_1 = (h_1 + a_{80}) \& 0xFFFFFFFF$

$h_2 = (h_2 + b_{80}) \& 0xFFFFFFFF$

$h_3 = (h_3 + c_{80}) \& 0xFFFFFFFF$

$h_4 = (h_4 + d_{80}) \& 0xFFFFFFFF$

$h_5 = (h_5 + e_{80}) \& 0xFFFFFFFF$

$\Re = h_1 + h_2 + h_3 + h_4 + h_5$

Aşağıdaki tabloda kaba kodda kullanılan sabitler verilmiştir.[22]

Tur	**Adımlar**	**Kullanılan f_t fonksiyonu**	**Kullanılan k_t sabiti**
1	**1-20**	$(x \wedge y) \vee (\neg x \wedge z)$	0x5a827999
2	**21-40**	$x \oplus y \oplus z$	0x6ed6eba1
3	**41-60**	$(x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$	0x8fabbcdc
4	**61-80**	$x \oplus y \oplus z$	0xca62c1d6

Tablo-5 SHA-1 Algoritması için Kullanılan Sabitler (Wang,X. , Yin,Y,L. , and Yu,H.)

SHA özetleme fonksiyonu grubunda, SHA-0 VE SHA-1 dışında özet değeri daha uzun olan, SHA-2 özetleme grubu;SHA-224, SHA-256, SHA-384 ve SHA-512 algoritmaları ile vardır. Bu algoritmaların özellikleri Tablo 6'da özetlenmiştir.

SHA-0 ve SHA-1 için geliştirilmiş ataklar SHA-2 versiyonları için herhangi bir zayıflık belirtmemiştir. Ancak SHA-2 algoritmasının da yakın zamanda kırılma ihtimallerine karşı araştırmacılar yeni algoritma tasarlamaya karar vermişlerdir. [24]

	Mesaj Uzunluğu	Blok Uzunluğu	Mesaj Özeti	Güvenlik
SHA1	< 264	512	160	80
SHA256	< 264	512	256	128
SHA384	< 2128	1024	384	192
SHA512	< 2128	1024	512	256

Tablo-6 SHA Fonksiyonlarının Karşılaştırılması (Çalık,Ç., Turan, M, S., Yüce,Z)

4.5 MD Özetleme Fonksiyonu

MD5 özetleme algoritması (RFC 1321) Ron Rivest tarafından geliştirilmiştir. Kaba kuvvet ve kriptoanaliz gelişmeden son birkaç yıla kadar MD5 en çok kullanılan güvenli özetleme algoritmasıydı. Algoritma giriş olarak sabit uzunlukta bir mesaj alır ve çıkış olarak 128 bitlik bir mesaj özeti üretir. Giriş 512 bloklar şeklinde işletilir. İşlemci hızı yükseldikçe 128 bitlik özetleme kodunun güvenliği de sarsıldı.[18]

4.6 RIPE-MD-160 Özetleme Algoritması

RIPEMD-160 MD4 ve MD5 üzerinde başarılı ataklar yapan Europen RACE Integrity Primitives Evaluation (RIPE) projesi altında birleşen bir grup araştırmacı tarafından geliştirmiştir. Grup aslında RIPEM algoritmasını 128 bitlik verisyonunu geliştirdi. RIPE projesi bittikten sonra H.Dobbertin RIPE-MD algoritmasının iki turu üzerinde ataklar geliştirdi .Bu ataklardan sonra RIPE konsursiyumun bazı üyeleri RIPEMD diye değiştirmeye karar verdiler. Tasarım çalışmaları RIPE projesi üyeleri ve H.Dobbertin tarafından yapıldı. RIPEMD-160 yapısal olarak SHA-1 e çok yakındır. Algoritma giriş olarak sabit uzunlukta bir mesaj alır ve çıkış olarak 160 bitlik mesaj özeti oluşturur. Giriş 512 bitlik bloklar şeklinde işletilir.[18],[12]. RIPEMD özetleme algoritması Avrupa Birliği direktifinde, e-imza oluşturmada kullanılması kabul görülen standart özetleme algoritmasıdır.

Aşağıdaki tabloda MD5, SHA-1 ve RIPEMD özetleme fonksiyonları karşılaştırılmıştır.

	MD5	**SHA-1**	**RIPEMD-160**
Uzunluk	128 bit	160 bit	160 bit
İşleme birimi	512 bit	512 bit	512 bit
Adım sayısı	64(16 adımlık 4 tur)	80 (20 adımlık 4 tur)	160 (16 adımlık 5 çift tur)
Max. Mesaj uzunluğu	∞	$2^{64}-1$ bit	∞
Temel mantık fonksiyon sayısı	4	4	5
Kullanılan sabit değerler sayısı	64	4	9

Tablo-7 Özetleme Algoritmalarının Karşılaştırlması (MD5,SHA-1,RIPEMD) (Stallings,W 1999)

4.7 Hash Fonksiyonlarına Yapılan Ataklar

Hash fonksiyonlarına yapılacak atakların anlamı; içerik olarak farklı olan farklı iki mesaj aynı özet değerinde çakıştırmaktır. Dolayısıyla bir özetleme fonksiyonun farklı mesajlar üzerinden alınan özet değerlerinin çakışmalarına karşı direnç göstermesi beklenir. "N elemanlı bir popülasyonda, bir çakışma gerçekleşebilmesi için ortalamada $\sqrt{n}$ tane rasgele örnekleme yapılması beklenmektedir. Dolayısıyla, $2^{n/2}$ lik karmaşıklıktan daha az işlemle çakışma bulan bir algoritma özet fonksiyonunu kırmış sayılır".[24] Diğer bir ifadeyle böyle bir çakışmanın gerçekleşmesi özet fonksiyonunun ürettiği özet değerinin yarı uzunluğuna bağlıdır.

Farklı içerikli iki mesajın, mesaj özetlerinin çakışması durumunda, mesaj sahibi, kendi imzaladığı mesajı inkar edebilir. Bu durum e-imza güvenliği açısından çok büyük risk olup, çakışma durumundaki imzalı mesaj verisinin geçerlilik durumunun iptaline sebep olur.

2005 yılında, SHA-1 fonksiyonunda çakışmaları 2^{63} işlemde bulan bir atak geliştirilmiştir. [24]. 2^{63} seviyesindeki işlem adımları fazla olsa da, günümüzde işlem hacmi ve hızı artan bilgisayarlar için makul zaman dilimlerinde gerçekleştirilebilir bir ataktır.

BEŞİNCİ BÖLÜM
SONUÇ ve DEĞERLENDİRMELER

Çağımızın bilgi çağı olması dolayısıyla, toplumların bilgiye en kısa ve en güvenli yoldan ulaşma gereksinimini zorunlu hale getirip, toplumsal gelişimin ön koşulu durumuna gelmiştir. Günümüzde pozitif yönde değişimin ve gelişimin ölçüsü olarak kabul görülen bilgi ve bilgiye bağlı gelişen bilgi teknolojileri, insanoğluna kolay kaliteli ve makul maliyetlerde yaşam standartları sunmaktadır. Dünyanın gelişmiş ülkelerinin gelişim seyri, sahip oldukları bilimsel ve teknolojik düzeyleriyle direkt orantılıdır.

Bilişim teknolojileri kapsamında, bilgisayar ve bilgisayar sistemlerinin baş döndürücü gelişimi ve bu gelişime paralel olarak yaşamsal faaliyetlerimize etkin bir biçimde girmesi, bu alandaki gelişmelerin yeterince dikkate alınması gerektiği sonucunu doğurur. Bilgisayar ve bilgisayar sistemleri arasındaki ağlar, hemen hemen bütün verilerin (ses, görüntü, data) zamandan ve mekandan bağımsız iletimini olanaklı hale getirmiştir. Bu durum e-devlet, e-ticaret uygulamalarının vazgeçilmez alt yapısıdır. Kağıtsız ofisler, zaman ve mekan kısıtlaması olmadan, kaliteli ve güvenli hizmet kalitesinin elde edilmesini sağlar.

İnternet üzerindeki bilgi dolaşımı verinin güvenliğini, bilginin varlığından daha önemli hale getirilmiştir. Zira açık ağlar üzerinde gezinen veri her an kötü niyetli kişilerin atağına maruz kalmak gibi bir ihtimalle karşı karşıyadır. Bilginin gönderen uçtan hedef uca değiştirilmeden, güvenli olarak gönderilmesi, İnternetin kolaylıklarından ve avantajlarından yeterince faydalanmanın önemli bir şartıdır. Çünkü güvenliğini sağlayamadığımız bilginin, kolay iletiminin hiçbir değeri yoktur.

Kriptoloji biliminin gelişmesiyle beraber, İnternet üzerinde dolaşan verinin gizliliği, kullanılan kriptolojik algoritmanın, gizliliği oluşturma niteliğiyle orantılı bir düzeyde sağlanması olanaklı hale gelmiştir. Çalışmanın birinci bölümünde kriptolojinin tarihçesi ve ayrıca üçüncü bölümde de simetrik ve asimetrik

kriptosistemlerinin, üzerinde inşa edildikleri matematiksel algoritmik yaklaşımlarla detaylandırılmıştır.

İnternet üzerinden gönderilen verilerin bir şifreleme algoritmasıyla gizliliği sağlandığında, verinin gönderici cephesinde kimlik bildirimi eksik kalır. Gerçekleştirilen işlemlerin kimlik tanımlanması yapılmamışsa hukuksal olarak geçerli değildir.

E-imzanın etkin kullanımıyla İnternet üzerinden yapılan iş ve işlemler teknik ve hukuksal olarak geçerlilik kazanır. E-imza uygulamaları kurumlar arası iletişim ağlarında, e-ticaret ve e-devlet dönüşümlerinde mutlaka kullanılmalıdır. Böylece kaliteli, hızlı ve güvenli iş akışıyla verimli sonuçlar elde edilmiş olur. Bu çalışmamızda e-imza uygulamalarının olumlu/verimli sonuçlarının toplum geneline yayılması için aşağıdaki öneriler sunulmuştur:

- E-ticaret ve e-devlet dönüşümlerinin gerçekleştirilmesi e-imza uygulamalarının etkin kullanılmasıyla, e-imza uygulamaları da bilgisayar sistemleri üzerinden gerçekleştirildiğinde, bilgisayar okur-yazarlığı yaygınlaştırılmalıdır.

- E-imza ile ilgili hukuksal geçerliliğin sağlanması açısından, hukuksal düzenlemeler bütün uygulamaları kapsayacak şekilde düzenlenmelidir.

- E-imzanın uygulanabilir alt yapısı olarak, teknik standartları yeterli "Açık Anahtar Alt Yapısı" oluşturulmalıdır.

- E-imza oluşturmak kullanılan asimetrik şifreleme algoritması olarak, RSA algoritmasıyla aynı güvenlik seviyesini, çok daha düşük bit seviyelerinde sağladığından "Eliptik Eğri" Algoritması kullanılmalıdır.(Burada kulanım amacı da önemlidir.)

- E-imza ile imzalanmış mesajın, mesaj özetinin SHA-2 veya RIPE-MD-160 ile oluşturulmalıdır. Çünkü SHA-0, SHA-1 ve MD serisindeki bir çok özetleme fonksiyonu kırılmış durumdadır.

- İmzalanacak mesaj içeriği, gizlilik sağlamak için simetrik şifreleme algoritmaları kullanılmalı, anahtar dağıtımı için asimetrik şifreleme algoritmaları kullanılmalıdır. Çünkü asimetrik şifreleme algoritmalarının işlem hacmi çok yüksek olduğundan, bellek karmaşıklığı ve işlem hızı açısından elverişli değildir.

- İleri seviyede güvenlik gerektiren uygulamalarda anahtar dağıtımı "Kuantum kriptografi " ile sağlanmalıdır. Çünkü fiber optik ağ üzerinde ilerleyen fotonla dağıtılan anahtarların dinlenmesi, bugünkü teknoloji ile mümkün değildir. Bu durum fotonun ilerlerken Heiseberg 'in belirsizlik ilkesine göre konumunun ve momentumunun aynı anda belirlenememesiyle açıklanır.

- Gelecekte e-imza uygulamalarının artması, e-imza işlemlerinin taşınabilir aygıtlarda da gerçekleştirilmesi sonucunu getirecektir. Dünyanın bir çok ülkesinde mobil imza olarak bilinen m-imza uygulaması vardır. Ülkemizde Turkcell GSM şirketinin m-imza uygulamaları son bir yıl içinde başlatmış durumdadır.

- E-imza uygulamaları ile iş ve işlem gerçekleştirme düzeyi toplumların bilişim teknolojilerindeki gelişmişlik düzeylerini de belirleyecektir. Çünkü hızlı, kaliteli, güvenli, kolay gerçeklenebilirlik, zamandan ve mekandan bağımsız işlem gerçekleştirebilme amacı, insanlığın ortak hedefidir. E-imza uygulaması bu amacın vazgeçilmez aracıdır.

KAYNAKÇA

[1] Püsküllüoğlu A. ”Türkçe Sözlük” , Genişletilmiş 5. Baskı/Ekim-/ISBN 975-6770-38-4, Doğan Yayınları, 2004

[2] Sağıroğlu, Ş. ve Alkan, M. ,”Her Yönüyle Elektronik İmza”, 1. Basım Ankara Grafiker Yayınları, ISBN:975-6355-23-9, 2005

[3]http://www.e-imza.gen.tr/index.php?Page=EImzaNedir&YaziNo=4-

(06.04.2007)

[4] 6/1/2005 tarihli ve 25692 sayılı Resmî Gazete,5070 Sayılı Elektronik İmza Kanunu

[5] http://www.ueimzas.gazi.edu.tr/index.php?id=poster (11.O4.2007)

[6] Çalık,Ç., Sönmez Turan, M., Yüce,Z., ”E-imzada SHA-1 Özetleme Algoritmasının Kullanımı” Ulusal Elektronik İmza Sempozyumu Bildiriler Kitabı, 7-8 Aralık, Ankara, s.167,168,168, 2006

[7] http://www.ueimzas.gazi.edu.tr/pdf/bildiri/59.pdf (14.04.2007)

[8] http://www.ueimzas.gazi.edu.tr/pdf/bildiri/22.pdf (17.04.2007)

[9] http://www.tk.gov.tr/eimza/doc/diger/eimza_bgs_taslak_raporuV1.2.pdf

(28.04.2007)

[10] 5070 Sayılı Elektronik İmza Kanunu, Kabul Tarihi:15.12004

[11] http://www.iam.metu.edu.tr/lectures/IntroCryp (11.05.2007)

[12] Stinson,D. “Crytography:Theory and Practice”, CRC Press, CRC Press LLC, Second Edition, ISBN:0849385210 Pub, 17/03/1995

[13] Joe Hurd, ”Computer Laboratory”, University of Cambridge Intel Corp., 11 Aralık 2006

[14]http://144.122.9.63/yuup/edonusum/20041218/AAA_arastirma_grubu_sunum.ppt (26.05.2007)

[15] http://fmd.ksu.edu.tr/sayi/81/81.35-40.pdf (23.05.2007)

[16] http://www.iam.metu.edu.tr/sempozyum/2005/sunumlar/051119_MCenk.pdf

(27.05.2007)

[17]http://www.certicom.com/download/aid-111/cert_ecc_challenge.pdf

(26.05.2007)

[18] Stallings,W. , “Network Security Essentials, Application and Standards ”, ISBN 0-13-016093-8,TK 5105.59.S725, 1999

[19] http://www.ueimzas.gazi.edu.tr/pdf/bildiri/25.pdf (27.05.2007)

[20] http://people.sabanciuniv.edu/levi/bilisim02.pdf (28.05.2007)

[21] http://www.ueimzas.gazi.edu.tr/pdf/bildiri/24.pdf (30.05.2007)

[22] X. Wang, Y. L. Yin, and H. Yu. “Finding Collisions in the Full SHA-1” Crypto, 2005
[23] http://www.ueimzas.gazi.edu.tr/pdf/poster/14.pdf (05.05.2007)
[24] http://www.ueimzas.gazi.edu.tr/pdf/poster/54.pdf

EKLER

Ek-1 Elektronik İmza Kanunu

ELEKTRONİK İMZA KANUNU

Kanun No.5070 **Kabul Tarihi: 15.01.2004**

BİRİNCİ KISIM

Amaç, Kapsam ve Tanımlar

Amaç

MADDE 1.- Bu Kanunun amacı, elektronik imzanın hukukî ve teknik yönleri ile kullanımına ilişkin esasları düzenlemektir.

Kapsam

MADDE 2.- Bu Kanun, elektronik imzanın hukukî yapısını, elektronik sertifika hizmet sağlayıcılarının faaliyetlerini ve her alanda elektronik imzanın kullanımına ilişkin işlemleri kapsar.

Tanımlar

MADDE 3.- Bu Kanunda geçen;

a) Elektronik veri: Elektronik, optik veya benzeri yollarla üretilen, taşınan veya saklanan kayıtları,

b) Elektronik imza: Başka bir elektronik veriye eklenen veya elektronik veriyle mantıksal bağlantısı bulunan ve kimlik doğrulama amacıyla kullanılan elektronik veriyi,

c) İmza sahibi: Elektronik imza oluşturmak amacıyla bir imza oluşturma aracını kullanan gerçek kişiyi,

d) İmza oluşturma verisi: İmza sahibine ait olan, imza sahibi tarafından elektronik imza oluşturma amacıyla kullanılan ve bir eşi daha olmayan şifreler, kriptografik gizli anahtarlar gibi verileri,

e) İmza oluşturma aracı: Elektronik imza oluşturmak üzere, imza oluşturma verisini kullanan yazılım veya donanım aracını,

f) İmza doğrulama verisi: Elektronik imzayı doğrulamak için kullanılan şifreler, kriptografik açık anahtarlar gibi verileri,

g) İmza doğrulama aracı: Elektronik imzayı doğrulamak amacıyla imza doğrulama verisini kullanan yazılım veya donanım aracını,

h) Zaman damgası: Bir elektronik verinin, üretildiği, değiştirildiği, gönderildiği, alındığı ve / veya kaydedildiği zamanın tespit edilmesi amacıyla, elektronik sertifika hizmet sağlayıcısı tarafından elektronik imzayla doğrulanan kaydı,

ı) Elektronik sertifika: İmza sahibinin imza doğrulama verisini ve kimlik bilgilerini birbirine bağlayan elektronik kaydı,

j) Kurum: Telekomünikasyon Kurumunu, İfade eder.

İKİNCİ KISIM
Güvenli Elektronik İmza ve Sertifika Hizmetleri

BİRİNCİ BÖLÜM
Güvenli Elektronik İmza, Güvenli Elektronik İmza Oluşturma ve Doğrulama Araçları

Güvenli elektronik imza

MADDE 4.- Güvenli elektronik imza;

a) Münhasıran imza sahibine bağlı olan,

b) Sadece imza sahibinin tasarrufunda bulunan güvenli elektronik imza oluşturma aracı ile oluşturulan,

c) Nitelikli elektronik sertifikaya dayanarak imza sahibinin kimliğinin tespitini sağlayan,

d) İmzalanmış elektronik veride sonradan herhangi bir değişiklik yapılıp yapılmadığının tespitini sağlayan,

Elektronik imzadır.

Güvenli elektronik imzanın hukukî sonucu ve uygulama alanı

MADDE 5.- Güvenli elektronik imza, elle atılan imza ile aynı hukukî sonucu doğurur.

Kanunların resmî şekle veya özel bir merasime tabi tuttuğu hukukî işlemler ile teminat sözleşmeleri güvenli elektronik imza ile gerçekleştirilemez.

Güvenli elektronik imza oluşturma araçları

MADDE 6.- Güvenli elektronik imza oluşturma araçları;

a) Ürettiği elektronik imza oluşturma verilerinin kendi aralarında bir eşi daha bulunmamasını,

b) Üzerinde kayıtlı olan elektronik imza oluşturma verilerinin araç dışına hiçbir biçimde çıkarılamamasını ve gizliliğini,

c) Üzerinde kayıtlı olan elektronik imza oluşturma verilerinin, üçüncü kişilerce elde edilememesini, kullanılamamasını ve elektronik imzanın sahteciliğe karşı korunmasını,

d) İmzalanacak verinin imza sahibi dışında değiştirilememesini ve bu verinin imza sahibi tarafından imzanın oluşturulmasından önce görülebilmesini,

Sağlayan imza oluşturma araçlarıdır.

Güvenli elektronik imza doğrulama araçları

MADDE 7.- Güvenli elektronik imza doğrulama araçları;

a) İmzanın doğrulanması için kullanılan verileri, değiştirmeksizin doğrulama yapan kişiye gösteren,

b) İmza doğrulama işlemini güvenilir ve kesin bir biçimde çalıştıran ve doğrulama sonuçlarını değiştirmeksizin doğrulama yapan kişiye gösteren,

c) Gerektiğinde, imzalanmış verinin güvenilir bir biçimde gösterilmesini sağlayan,

d) İmzanın doğrulanması için kullanılan elektronik sertifikanın doğruluğunu ve geçerliliğini güvenilir bir biçimde tespit ederek sonuçlarını değiştirmeksizin doğrulama yapan kişiye gösteren,

e) İmza sahibinin kimliğini değiştirmeksizin doğrulama yapan kişiye gösteren,

f) İmzanın doğrulanması ile ilgili şartlara etki edecek değişikliklerin tespit edilebilmesini sağlayan,

İmza doğrulama araçlarıdır.

İKİNCİ BÖLÜM

Elektronik Sertifika Hizmet Sağlayıcısı, Nitelikli Elektronik Sertifika ve Yabancı Elektronik Sertifikalar

Elektronik sertifika hizmet sağlayıcısı

MADDE 8.- Elektronik sertifika hizmet sağlayıcısı, elektronik sertifika, zaman damgası ve elektronik imzalarla ilgili hizmetleri sağlayan kamu kurum ve kuruluşları ile gerçek veya özel hukuk tüzel kişilerdir. Elektronik sertifika hizmet sağlayıcısı, Kuruma yapacağı bildirimden iki ay sonra faaliyete geçer.

Elektronik sertifika hizmet sağlayıcısı yapacağı bildirimde;

a) Güvenli ürün ve sistemleri kullanmak,

b) Hizmeti güvenilir bir biçimde yürütmek,

c) Sertifikaların taklit ve tahrif edilmesini önlemekle ilgili her türlü tedbiri almak,

İle ilgili şartları sağladığını ayrıntılı bir biçimde gösterir.

Kurum, yukarıdaki şartlardan birinin eksikliğini veya yerine getirilmediğini tespit ederse, bu eksikliklerin giderilmesi için, elektronik sertifika hizmet sağlayıcısına bir ayı geçmemek üzere bir süre verir, bu süre içinde elektronik sertifika hizmet sağlayıcısının faaliyetlerini durdurur. Sürenin sonunda eksikliklerin giderilmemesi halinde elektronik sertifika hizmet sağlayıcısının faaliyetine son verir. Kurumun bu kararlarına karşı 19 uncu maddenin ikinci fıkrası hükümleri gereğince itiraz edilebilir.

Elektronik sertifika hizmet sağlayıcılarının faaliyetlerinin devamı sırasında bu maddede gösterilen şartları kaybetmeleri hâlinde de yukarıdaki fıkra hükümleri uygulanır.

Elektronik sertifika hizmet sağlayıcıları, Kurumun belirleyeceği ücret alt ve üst sınırlarına uymak zorundadır.

Nitelikli elektronik sertifika

MADDE 9.- Nitelikli elektronik sertifikada;

a) Sertifikanın "nitelikli elektronik sertifika" olduğuna dair bir ibarenin,

b) Sertifika hizmet sağlayıcısının kimlik bilgileri ve kurulduğu ülke adının,

c) İmza sahibinin teşhis edilebileceği kimlik bilgilerinin,

d) Elektronik imza oluşturma verisine karşılık gelen imza doğrulama verisinin,

e) Sertifikanın geçerlilik süresinin başlangıç ve bitiş tarihlerinin,

f) Sertifikanın seri numarasının,

g) Sertifika sahibi diğer bir kişi adına hareket ediyorsa bu yetkisine ilişkin bilginin,

h) Sertifika sahibi talep ederse meslekî veya diğer kişisel bilgilerinin,

ı) Varsa sertifikanın kullanım şartları ve kullanılacağı işlemlerdeki maddî sınırlamalara ilişkin bilgilerin,

j) Sertifika hizmet sağlayıcısının sertifikada yer alan bilgileri doğrulayan güvenli elektronik imzasının,

Bulunması zorunludur.

Elektronik sertifika hizmet sağlayıcısının yükümlülükleri

MADDE 10.- Elektronik sertifika hizmet sağlayıcısı;

a) Hizmetin gerektirdiği nitelikte personel istihdam etmekle,

b) Nitelikli sertifika verdiği kişilerin kimliğini resmî belgelere göre güvenilir bir biçimde tespit etmekle,

c) Sertifika sahibinin diğer bir kişi adına hareket edebilme yetkisi, meslekî veya diğer kişisel bilgilerinin sertifikada bulunması durumunda, bu bilgileri de resmî belgelere dayandırarak güvenilir bir biçimde belirlemekle,

d) İmza oluşturma verisinin sertifika hizmet sağlayıcısı tarafından veya sertifika talep eden kişi tarafından sertifika hizmet sağlayıcısına ait yerlerde üretilmesi durumunda bu işlemin gizliliğini sağlamak veya sertifika hizmet sağlayıcısının sağladığı araçlarla üretilmesi durumunda, bu işleyişin güvenliğini sağlamakla,

e) Sertifikanın kullanımına ilişkin özelliklerin ve uyuşmazlıkların çözüm yolları ile ilgili şartların ve kanunlarda öngörülen sınırlamalar saklı kalmak üzere güvenli elektronik imzanın elle atılan imza ile eşdeğer olduğu hakkında sertifika talep eden kişiyi sertifikanın tesliminden önce yazılı olarak bilgilendirmekle,

f) Sertifikada bulunan imza doğrulama verisine karşılık gelen imza oluşturma verisini başkasına kullandırmaması konusunda, sertifika sahibini yazılı olarak uyarmak ve bilgilendirmekle,

g)Yaptığı hizmetlere ilişkin tüm kayıtları yönetmelikle belirlenen süreyle saklamakla,

h) Faaliyetine son vereceği tarihten en az üç ay önce durumu Kuruma ve elektronik sertifika sahibine bildirmekle,

Yükümlüdür.

Elektronik sertifika hizmet sağlayıcısı üretilen imza oluşturma verisinin bir kopyasını alamaz veya bu veriyi saklayamaz.

Nitelikli elektronik sertifikaların iptal edilmesi

MADDE 11.- Elektronik sertifika hizmet sağlayıcısı;

a) Nitelikli elektronik sertifika sahibinin talebi,

b) Sağladığı nitelikli elektronik sertifikaya ilişkin veri tabanında bulunan bilgilerin sahteliğinin veya yanlışlığının ortaya çıkması veya bilgilerin değişmesi,

c) Nitelikli elektronik sertifika sahibinin fiil ehliyetinin sınırlandığının, iflâsının veya gaipliğinin ya da ölümünün öğrenilmesi,

Durumunda vermiş olduğu nitelikli elektronik sertifikaları derhâl iptal eder.

Elektronik sertifika hizmet sağlayıcısı, nitelikli elektronik sertifikaların iptal edildiği zamanın tam olarak tespit edilmesine imkân veren ve üçüncü kişilerin hızlı ve güvenli bir biçimde ulaşabileceği bir kayıt oluşturur.

Elektronik sertifika hizmet sağlayıcısı, faaliyetine son vermesi ve vermiş olduğu nitelikli elektronik sertifikaların başka bir elektronik sertifika hizmet sağlayıcısı tarafından kullanımının sağlanamaması durumunda vermiş olduğu nitelikli elektronik sertifikaları derhâl iptal eder.

Elektronik sertifika hizmet sağlayıcısının faaliyetine Kurum tarafından son verilmesi halinde Kurum, faaliyetine son verilen elektronik sertifika hizmet sağlayıcısının vermiş olduğu nitelikli elektronik sertifikaların başka bir elektronik sertifika hizmet sağlayıcısına devredilmesine karar verir ve durumu ilgililere duyurur.

Elektronik sertifika hizmet sağlayıcısı geçmişe yönelik olarak nitelikli elektronik sertifika iptal edemez.

Bilgilerin korunması

MADDE 12.- Elektronik sertifika hizmet sağlayıcısı;

a) Elektronik sertifika talep eden kişiden, elektronik sertifika vermek için gerekli bilgiler hariç bilgi talep edemez ve bu bilgileri kişinin rızası dışında elde edemez,

b) Elektronik sertifika sahibinin izni olmaksızın sertifikayı üçüncü kişilerin ulaşabileceği ortamlarda bulunduramaz,

c) Elektronik sertifika talep eden kişinin yazılı rızası olmaksızın üçüncü kişilerin kişisel verileri elde etmesini engeller. Bu bilgileri sertifika sahibinin onayı olmaksızın üçüncü kişilere iletemez ve başka amaçlarla kullanamaz.

Hukukî sorumluluk

MADDE 13.- Elektronik sertifika hizmet sağlayıcısının, elektronik sertifika sahibine karşı sorumluluğu genel hükümlere tâbidir.

Elektronik sertifika hizmet sağlayıcısı, bu Kanun veya bu Kanuna dayanılarak çıkarılan yönetmelik hükümlerinin ihlâli suretiyle üçüncü kişilere verdiği zararları tazminle yükümlüdür. Elektronik sertifika hizmet sağlayıcısı kusursuzluğunu ispat ettiği takdirde tazminat ödeme yükümlülüğü doğmaz.

Elektronik sertifika hizmet sağlayıcısı, söz konusu yükümlülük ihlâlinin istihdam ettiği kişilerin davranışına dayanması hâlinde de zarardan sorumlu olup, elektronik sertifika hizmet sağlayıcısı, bu sorumluluğundan, Borçlar Kanununun 55 inci maddesinde öngörülen türden bir kurtuluş kanıtı getirerek kurtulamaz.

Nitelikli elektronik sertifikanın içerdiği kullanım ve maddî kapsamına ilişkin sınırlamalar hariç olmak üzere, elektronik sertifika hizmet sağlayıcısının üçüncü

kişilere ve nitelikli elektronik imza sahibine karşı sorumluluğunu ortadan kaldıran veya sınırlandıran her türlü şart geçersizdir.

Elektronik sertifika hizmet sağlayıcısı, bu Kanundan doğan yükümlülüklerini yerine getirmemesi sonucu doğan zararların karşılanması amacıyla sertifika malî sorumluluk sigortası yaptırmak zorundadır. Sigortaya ilişkin usul ve esaslar Hazine Müsteşarlığının görüşü alınarak Kurum tarafından çıkarılacak yönetmelikle belirlenir.

Bu maddede öngörülen sertifika malî sorumluluk sigortası Türkiye'de ilgili branşta çalışmaya yetkili olan sigorta şirketleri tarafından yapılır. Bu sigorta şirketleri sertifika malî sorumluluk sigortasını yapmakla yükümlüdürler. Bu yükümlülüğe uymayan sigorta şirketlerine Hazine Müsteşarlığınca sekizmilyar lira idarî para cezası verilir. Bu para cezasının tahsilinde ve cezaya itiraz usulünde 18 inci madde hükümleri uygulanır.

Elektronik sertifika hizmet sağlayıcısı, nitelikli elektronik sertifikayı elektronik imza sahibine sigorta ettirerek teslim etmekle yükümlüdür.

Yabancı elektronik sertifikalar

MADDE 14.- Yabancı bir ülkede kurulu bir elektronik sertifika hizmet sağlayıcısı tarafından verilen elektronik sertifikaların hukukî sonuçları milletlerarası anlaşmalarla belirlenir.

Yabancı bir ülkede kurulu bir elektronik sertifika hizmet sağlayıcısı tarafından verilen elektronik sertifikaların, Türkiye'de kurulu bir elektronik sertifika hizmet sağlayıcısı tarafından kabul edilmesi durumunda, bu elektronik sertifikalar nitelikli elektronik sertifika sayılır. Bu elektronik sertifikaların kullanılması sonucunda doğacak zararlardan, Türkiye'deki elektronik sertifika hizmet sağlayıcısı da sorumludur.

ÜÇÜNCÜ KISIM

Denetim ve Ceza Hükümleri

Denetim

MADDE 15.- Elektronik sertifika hizmet sağlayıcılarının bu Kanunun uygulanmasına ilişkin faaliyet ve işlemlerinin denetimi Kurumca yerine getirilir.

Kurum, gerekli gördüğü zamanlarda elektronik sertifika hizmet sağlayıcılarını denetleyebilir. Denetleme sırasında, denetleme yapmaya yetkili görevliler tarafından her türlü defter, belge ve kayıtların verilmesi, yönetim yerleri, binalar ve eklentilerine girme, yazılı ve sözlü bilgi alma, örnek alma ve işlem ve hesapları

denetleme isteminin elektronik sertifika hizmet sağlayıcıları ve ilgililer tarafından yerine getirilmesi zorunludur.

İmza oluşturma verilerinin izinsiz kullanımı

MADDE 16.- Elektronik imza oluşturma amacı ile ilgili kişinin rızası dışında; imza oluşturma verisi veya imza oluşturma aracını elde eden, veren, kopyalayan ve bu araçları yeniden oluşturanlar ile izinsiz elde edilen imza oluşturma araçlarını kullanarak izinsiz elektronik imza oluşturanlar bir yıldan üç yıla kadar hapis ve beşyüz milyon liradan aşağı olmamak üzere ağır para cezasıyla cezalandırılırlar.

Yukarıdaki fıkrada işlenen suçlar elektronik sertifika hizmet sağlayıcısı çalışanları tarafından işlenirse bu cezalar yarısına kadar artırılır.

Bu maddedeki suçlar nedeniyle oluşan zarar ayrıca tazmin ettirilir.

Elektronik sertifikalarda sahtekârlık

MADDE 17.- Tamamen veya kısmen sahte elektronik sertifika oluşturanlar veya geçerli olarak oluşturulan elektronik sertifikaları taklit veya tahrif edenler ile yetkisi olmadan elektronik sertifika oluşturanlar veya bu elektronik sertifikaları bilerek kullananlar, fiilleri başka bir suç oluştursa bile ayrıca, iki yıldan beş yıla kadar hapis ve birmilyar liradan aşağı olmamak üzere ağır para cezasıyla cezalandırılırlar.

Yukarıdaki fıkrada işlenen suçlar elektronik sertifika hizmet sağlayıcısı çalışanları tarafından işlenirse bu cezalar yarısına kadar artırılır.

Bu maddedeki suçlar nedeniyle oluşan zarar ayrıca tazmin ettirilir.

İdarî para cezaları

MADDE 18.- Bu Kanunun;

a) 10 uncu maddesindeki yükümlülüklerinden herhangi birini yerine getirmeyen elektronik sertifika hizmet sağlayıcısına onmilyar lira,

b) 11 inci maddesindeki yükümlülüklerden herhangi birini yerine getirmeyen elektronik sertifika hizmet sağlayıcısına sekizmilyar lira,

c) 12 nci maddesi hükümlerine aykırı hareket edenler hakkında onmilyar lira,

d) 13 üncü maddesinin beş ve yedinci fıkralarındaki yükümlülükleri yerine getirmeyen elektronik sertifika hizmet sağlayıcısına sekizmilyar lira,

e) 15 inci maddesi hükmüne aykırı hareket eden elektronik sertifika hizmet sağlayıcısına yirmimilyar lira,

İdarî para cezası Telekomünikasyon Kurulu tarafından verilir. Verilen para cezalarına dair kararlar ilgililere 7201 sayılı Tebligat Kanunu hükümlerine göre tebliğ edilir. Bu cezalara karşı tebliğ tarihinden itibaren en geç yedi gün içinde

yetkili idare mahkemesine itiraz edilebilir. İtiraz, verilen cezanın yerine getirilmesini durdurmaz. İtiraz, zaruret görülmeyen hâllerde, evrak üzerinden inceleme yapılarak en kısa sürede sonuçlandırılır. İtiraz üzerine verilen kararlara karşı Bölge İdare Mahkemesine başvurulabilir. Bölge İdare Mahkemesinin verdiği kararlar kesindir. Bu Kanuna göre verilen idarî para cezaları, Kurumun bildirimi üzerine 6183 sayılı Amme Alacaklarının Tahsil Usulü Hakkında Kanun hükümlerine göre Maliye Bakanlığınca tahsil olunur.

İdarî nitelikteki suçların tekrarı ve kapatma

MADDE 19.- 18 inci maddedeki suçları işleyenlerin bu suçları işledikleri tarihten itibaren geriye doğru üç yıl içinde ikinci kez işlemeleri hâlinde para cezaları iki kat olarak uygulanır, üçüncü kez işlemeleri hâlinde ise Kurum tarafından elektronik sertifika hizmet sağlayıcıları hakkında kapatma cezası verilir.

Kapatma cezası verilmesine ilişkin karar 7201 sayılı Tebligat Kanununa göre ilgililere tebliğ edilir. Bu karara karşı tebliğ tarihinden itibaren en geç yedi gün içinde yetkili idare mahkemesine itiraz edilebilir. İtiraz, yetkili makam tarafından verilen kapatma kararının yerine getirilmesini durdurmaz. İtiraz, zaruret görülmeyen hâllerde, evrak üzerinden inceleme yapılarak en kısa sürede sonuçlandırılır. İtiraz üzerine verilen kararlara karşı Bölge İdare Mahkemesine başvurulabilir. Bölge İdare Mahkemesinin verdiği kararlar kesindir.

DÖRDÜNCÜ KISIM

Çeşitli Hükümler

Yönetmelik

MADDE 20.- Bu Kanunun 6, 7, 8, 10, 11 ve 14 üncü maddelerinin uygulanmasına ilişkin usul ve esaslar, Kanunun yürürlük tarihinden itibaren altı ay içinde ilgili kurum ve kuruluşların görüşleri alınarak Kurum tarafından çıkarılacak yönetmeliklerle düzenlenir.

Kamu kurum ve kuruluşları hakkında uygulanmayacak hükümler

MADDE 21.- Bu Kanunun 8 inci maddesinin dört ve beşinci fıkraları ile 15 ve 19 uncu maddesi hükümleri, elektronik sertifika hizmet sağlama faaliyeti yerine getiren kamu kurum ve kuruluşları hakkında uygulanmaz.

MADDE 22.- 22.4.1926 tarihli ve 818 sayılı Borçlar Kanununun 14 üncü maddesinin birinci fıkrasına aşağıdaki cümle eklenmiştir.

Güvenli elektronik imza elle atılan imza ile aynı ispat gücünü haizdir.

MADDE 23.- 18.6.1927 tarihli ve 1086 sayılı Hukuk Usulü Muhakemeleri Kanununa 295 inci maddeden sonra gelmek üzere aşağıdaki 295/A maddesi eklenmiştir.

MADDE 295/A- Usulüne göre güvenli elektronik imza ile oluşturulan elektronik veriler senet hükmündedir. Bu veriler aksi ispat edilinceye kadar kesin delil sayılırlar.

Dava sırasında bir taraf kendisine karşı ileri sürülen ve güvenli elektronik imza ile oluşturulmuş veriyi inkâr ederse, bu Kanunun 308 inci maddesi kıyas yoluyla uygulanır.

MADDE 24.- 5.4.1983 tarihli ve 2813 sayılı Telsiz Kanununun 7 nci maddesinin birinci fıkrasına aşağıdaki (m) bendi eklenmiş ve mevcut (m) bendi (n) bendi olarak teselsül ettirilmiştir.

m) Elektronik İmza Kanunu ile verilen görevleri yerine getirmek,

Yürürlük

MADDE 25.- Bu Kanun yayımı tarihinden altı ay sonra yürürlüğe girer.

Yürütme

MADDE 26.- Bu Kanun hükümlerini Bakanlar Kurulu yürütür.

Ek-2 Elektronik İmza ile İlgili Süreçlere ve Teknik Kriterlere İlişkin Tebliğ'de Değişiklik Yapılmasına Dair Tebliğ

Telekomünikasyon Kurumundan:

Madde 1 — 6/1/2005 tarihli ve 25692 sayılı Resmî Gazete'de yayımlanan Elektronik İmza ile İlgili Süreçlere ve Teknik Kriterlere İlişkin Tebliğ'in Algoritmalar ve Parametreler" başlıklı 6 ncı maddesi aşağıdaki şekilde değiştirilmiştir:

Algoritmalar ve Parametreler

Madde 6 — İmza oluşturma ve doğrulama verileri ile özetleme algoritmaları, ETSI TS 102 176-1 standardına ve aşağıda yer alan şartlara uygun olmalıdır:

a) İmza sahibinin imza oluşturma ve doğrulama verileri

i. RSA için en az 1024 bit veya

ii. DSA için en az 1024 bit veya

iii. DSA Eliptik Eğrisi için en az 163 bit

b) ESHS'nin imza oluşturma ve doğrulama verileri

i. RSA için en az 2048 bit veya

ii. DSA için en az 2048 bit veya

iii. DSA Eliptik Eğrisi için en az 256 bit

c) Özetleme algoritması

i. RIPEMD – 160 veya

ii. SHA – 1 veya

iii. SHA-224 veya

iv. SHA-256 veya

v. WHIRLPOOL

Yukarıda belirtilen algoritmalar ve parametreler 31/12/2008 tarihine kadar geçerlidir.

Madde 2 — Bu Tebliğ yayımı tarihinde yürürlüğe girer.

Madde 3 — Bu Tebliğ hükümlerini Telekomünikasyon Kurulu Başkanı yürütür.

Ek-3 SHA-1 Özetleme fonksiyonu için Kodlama

```
//******************************************************
// Ana Unit
//******************************************************

unit Main;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, SHA1, StdCtrls, Menus;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Edit2: TEdit;
    MainMenu1: TMainMenu;
    Dosya: TMenuItem;
    Do1: TMenuItem;
    N1: TMenuItem;
    k1: TMenuItem;
    OpenDialog1: TOpenDialog;
    Memo1: TMemo;
    Button2: TButton;
    Label1: TLabel;
    N2: TMenuItem;
    Hash1: TMenuItem;
    procedure Button1Click(Sender: TObject);
    procedure k1Click(Sender: TObject);
    procedure Do1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Hash1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
  edit2.Text := SHA1SelfTest(memo1.Text)
```

```
end;

procedure TForm1.k1Click(Sender: TObject);
begin
  Close;
end;

procedure TForm1.Do1Click(Sender: TObject);
begin
  if OpenDialog1.Execute then
    Memo1.Lines.LoadFromFile(OpenDialog1.FileName);
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  Close
end;

procedure TForm1.Hash1Click(Sender: TObject);
begin
  Button1Click(Self);
end;

END.


//*********************************************************
// SHA1
//*********************************************************

unit SHA1;

interface
uses
  Sysutils, Tools;

type
  TSHA1Digest= array[0..19] of byte;
  TSHA1Context= record
    Hash: array[0..4] of DWord;
    Hi, Lo: integer;
    Buffer: array[0..63] of byte;
    Index: integer;
  end;

function SHA1SelfTest(sVal: String): String;
procedure SHA1Init(var Context: TSHA1Context);
procedure SHA1Update(var Context: TSHA1Context; Buffer: pointer; Len: integer);
procedure SHA1Final(var Context: TSHA1Context; var Digest: TSHA1Digest);
```

```
//*****************************************************************
********
implementation
{$R-}

function SHA1SelfTest(sVal: String): String;
var
  Context: TSHA1Context;
  Digest: TSHA1Digest;
  sResult: String;
  iVal: integer;
begin
  SHA1Init(Context);
  SHA1Update(Context,@sVal[1],length(sVal));
  SHA1Final(Context,Digest);
  sResult := '';
  for iVal := 0 to 19 do
    sResult := sResult + Chr(Digest[iVal]);
  Result := sResult;
end;

//*****************************************************************
********
function F1(x, y, z: DWord): DWord;
begin
  Result:= z xor (x and (y xor z));
end;
function F2(x, y, z: DWord): DWord;
begin
  Result:= x xor y xor z;
end;
function F3(x, y, z: DWord): DWord;
begin
  Result:= (x and y) or (z and (x or y));
end;

//*****************************************************************
********
function RB(A: DWord): DWord;
begin
  Result:= (A shr 24) or ((A shr 8) and $FF00) or ((A shl 8) and $FF0000) or (A shl
24);
end;

procedure SHA1Compress(var Data: TSHA1Context);
var
  A, B, C, D, E, T: DWord;
  W: array[0..79] of DWord;
  i: integer;
begin
  Move(Data.Buffer,W,Sizeof(Data.Buffer));
```

```
  for i:= 0 to 15 do
    W[i]:= RB(W[i]);
  for i:= 16 to 79 do
    W[i]:= LRot32(W[i-3] xor W[i-8] xor W[i-14] xor W[i-16],1);
  A:= Data.Hash[0]; B:= Data.Hash[1]; C:= Data.Hash[2]; D:= Data.Hash[3]; E:=
Data.Hash[4];
  for i:= 0 to 19 do
  begin
    T:= LRot32(A,5) + F1(B,C,D) + E + W[i] + $5A827999;
    E:= D; D:= C; C:= LRot32(B,30); B:= A; A:= T;
  end;
  for i:= 20 to 39 do
  begin
    T:= LRot32(A,5) + F2(B,C,D) + E + W[i] + $6ED9EBA1;
    E:= D; D:= C; C:= LRot32(B,30); B:= A; A:= T;
  end;
  for i:= 40 to 59 do
  begin
    T:= LRot32(A,5) + F3(B,C,D) + E + W[i] + $8F1BBCDC;
    E:= D; D:= C; C:= LRot32(B,30); B:= A; A:= T;
  end;
  for i:= 60 to 79 do
  begin
    T:= LRot32(A,5) + F2(B,C,D) + E + W[i] + $CA62C1D6;
    E:= D; D:= C; C:= LRot32(B,30); B:= A; A:= T;
  end;
  Data.Hash[0]:= Data.Hash[0] + A;
  Data.Hash[1]:= Data.Hash[1] + B;
  Data.Hash[2]:= Data.Hash[2] + C;
  Data.Hash[3]:= Data.Hash[3] + D;
  Data.Hash[4]:= Data.Hash[4] + E;
  FillChar(W,Sizeof(W),0);
  FillChar(Data.Buffer,Sizeof(Data.Buffer),0);
end;

//*****************************************************************
********procedure SHA1Init(var Context: TSHA1Context);
begin
  Context.Hi:= 0; Context.Lo:= 0;
  Context.Index:= 0;
  FillChar(Context.Buffer,Sizeof(Context.Buffer),0);
  Context.Hash[0]:= $67452301;
  Context.Hash[1]:= $EFCDAB89;
  Context.Hash[2]:= $98BADCFE;
  Context.Hash[3]:= $10325476;
  Context.Hash[4]:= $C3D2E1F0;
end;

//*****************************************************************
********
procedure SHA1UpdateLen(var Context: TSHA1Context; Len: integer);
```

```
var
  i, k: integer;
begin
  for k:= 0 to 7 do
  begin
    i:= Context.Lo;
    Inc(Context.Lo,Len);
    if Context.Lo< i then
      Inc(Context.Hi);
  end;
end;

//****************************************************************
********
procedure SHA1Update(var Context: TSHA1Context; Buffer: pointer; Len: integer);
type
  PByte= ^Byte;
begin
  SHA1UpdateLen(Context,Len);
  while Len> 0 do
  begin
    Context.Buffer[Context.Index]:= PByte(Buffer)^;
    Inc(PByte(Buffer));
    Inc(Context.Index);
    Dec(Len);
    if Context.Index= 64 then
    begin
      Context.Index:= 0;
      SHA1Compress(Context);
    end;
  end;
end;

//****************************************************************
********
procedure SHA1Final(var Context: TSHA1Context; var Digest: TSHA1Digest);
type
  PDWord= ^DWord;
begin
  Context.Buffer[Context.Index]:= $80;
  if Context.Index>= 56 then
    SHA1Compress(Context);
  PDWord(@Context.Buffer[56])^:= RB(Context.Hi);
  PDWord(@Context.Buffer[60])^:= RB(Context.Lo);
  SHA1Compress(Context);
  Context.Hash[0]:= RB(Context.Hash[0]);
  Context.Hash[1]:= RB(Context.Hash[1]);
  Context.Hash[2]:= RB(Context.Hash[2]);
  Context.Hash[3]:= RB(Context.Hash[3]);
  Context.Hash[4]:= RB(Context.Hash[4]);
  Move(Context.Hash,Digest,Sizeof(Digest));
```

```
  FillChar(Context,Sizeof(Context),0);
end;


END.
//*********************************************************
// Yardımcı Tools Uniti
//*********************************************************
unit Tools;

interface
uses
  Sysutils;

type
{$IFDEF VER120}
  dword= longword;
{$ELSE}
  dword= longint;
{$ENDIF}

function LRot16(X: word; c: integer): word; assembler;
function RRot16(X: word; c: integer): word; assembler;
function LRot32(X: dword; c: integer): dword; assembler;
function RRot32(X: dword; c: integer): dword; assembler;
procedure XorBlock(I1, I2, O1: PByteArray; Len: integer);
procedure IncBlock(P: PByteArray; Len: integer);

implementation

function LRot16(X: word; c: integer): word; assembler;
asm
  mov ecx,&c
  mov ax,&X
  rol ax,cl
  mov &Result,ax
end;

function RRot16(X: word; c: integer): word; assembler;
asm
  mov ecx,&c
  mov ax,&X
  ror ax,cl
  mov &Result,ax
end;

function LRot32(X: dword; c: integer): dword; register; assembler;
asm
  mov ecx, edx
  rol eax, cl
end;
```

```
function RRot32(X: dword; c: integer): dword; register; assembler;
asm
  mov ecx, edx
  ror eax, cl
end;

procedure XorBlock(I1, I2, O1: PByteArray; Len: integer);
var
  i: integer;
begin
  for i:= 0 to Len-1 do
    O1[i]:= I1[i] xor I2[i];
end;

procedure IncBlock(P: PByteArray; Len: integer);
begin
  Inc(P[Len-1]);
  if (P[Len-1]= 0) and (Len> 1) then
    IncBlock(P,Len-1);
end;

END.
```