Selçuk-Teknik Dergisi *ISSN 1302-6178* *Journal of Selcuk-Technic*
Cilt 9, Sayı:1-2010 *Volume 9, Number:1-2010*

SİMETRİK VE ASİMETRİK ŞİFRELEME ALGORİTMALARININ KARŞILAŞTIRILMASI

Halife KODAZ[a]* , Fatih M. BOTSALI[b]

[a]Bilgisayar Mühendisliği Bölümü, Selçuk Üniversitesi, Alaeddin Keykubad Kampüsü, Konya
[b]Makine Mühendisliği Bölümü, Selçuk Üniversitesi, Alaeddin Keykubad Kampüsü, Konya

Özet

Şifreleme bilgisayar ağlarında haberleşme güvenliğini sağlamak için kullanılmaktadır. Bu nedenle günümüzde bilgisayarlarda ya da bilgisayar ağlarında şifrelemenin önemi gün geçtikçe artmaktadır. Bu çalışmada simetrik şifreleme algoritmaları ve asimetrik şifreleme algoritmaları hakkında bilgi verildikten sonra şifrelemede kullanılan anahtar boyutlarının analizi gerçekleştirilmiştir. Ayrıca şifreleme algoritmalarının performans kriterleri incelenmiştir.

Anahtar Kelimeler: şifreleme, açık anahtar, özel anahtar, kriptografi

COMPARISON OF SYMMETRIC AND ASYMMETRIC ENCRYPTION ALGORITHMS

Abstract

Encryption has been used for providing communication security in computer networks. Therefore, importance of encryption has been increased significantly in recent years. In this study, analysis of key bits used in encryption has been examined. Also, information about symmetric and asymmetric encryption algorithms have been

* Corresponding author. Tel.:+ 903322233711 Fax: +903322410635
E-mail address: hkodaz@selcuk.edu.tr (Halife KODAZ)

surveyed. Finally, performance criteria of cryptography algorithms have been investigated.

Keywords: encryption, public key, private key, cryptography

1. Giriş

Son yıllarda internet kullanımının yaygınlaşması bir takım güvenlik sorunlarını da beraberinde getirmiştir. Bunun başlıca sebepleri, internet'in açık bir sistem olması ve üzerinde dolaşan verinin gasp edilmeye uygun olmasıdır. İnternette alınan ve gönderilen veri paketleri birçok halka açık ağlardan geçer, bu da bu paketlere herkes tarafından ulaşmayı mümkün kılmaktadır. Son derece gizli bilgilerin internette dolaşması, önemli bir kaygı halini almaktadır [1-3]. Bu tür bilgileri korumak mümkün olmadıkça, internette iş yapmak veya gizli, şahsi yazışmalarda bulunmak asla güvenli olmayacaktır. Bilgi güvenliği; başkası tarafından dinlenme, bilginin değiştirilmesi, kimlik taklidi gibi tehditlerin ortadan kaldırılması ile sağlanmaktadır. Bilgi güvenliği sağlamada kullanılan temel araç kriptografidir. Kriptografi bilgi güvenliğini inceleyen ve anlaşılabileni anlaşılamaz yapan bir bilim dalıdır. Gizlilik, güvenilirlik, veri bütünlüğü, kimlik doğrulama, özgünlük ve inkâr edilemezlik gibi konular kriptografinin önemli çalışma alanlarıdır [1-4].

2. Şifreleme Algoritmaları

Bir göndericinin alıcıya açık ağlar üzerinden bir ileti göndermek istediği zaman, açık ağlardan gönderilen iletiler üçüncü şahıslar tarafından dinlenme ve değiştirilme tehdidi altındadır. Burada söz konusu ileti düz metindir. Bazı kullanımlarda *plaintext* adı da verilir [1, 6]. Bir iletinin içeriğini saklamak üzere yapılan gizleme işlemi de

şifrelemedir (*encryption*). Bu işlem düz metini anahtar kullanarak şifreli metine dönüştürmektedir. Bilginin içeriği başkalarının anlamayacağı hale gelmektedir. Bu bilgi bir yere iletilmek amacıyla şifrelenen bir mesaj veya saklanmak amacıyla şifrelenen bir bilgi olabilir. Şifrelenmiş bir ileti şifreli metindir (*ciphertext*). Şifreli metini düz metine geri çevirme işlemi şifre çözümüdür (*decrypt*). Şekil 1'de bu süreç gösterilmiştir.

Şekil 1 Şifreleme ve şifreyi çözme işlemleri

Şifreleme algoritmaları anahtar kullanma yöntemlerine göre genel olarak iki kategoriye ayrılmaktadır. Bunlar:

- Gizli anahtarlı (Simetrik) şifreleme algoritmaları
- Açık anahtarlı (Asimetrik) şifreleme algoritmaları

2.1 Simetrik şifreleme algoritmaları

Simetrik şifreleme algoritmaları şifreleme ve şifre çözme işlemleri için tek bir gizli anahtar kullanmaktadır. Bu durum veri şifreleme için matematiksel açıdan daha az problem çıkaran bir yaklaşımdır ve çok kullanılan bir yöntemdir. Bu tip algoritmalarda şifreleme işlemi gerçekleştirildikten sonra şifreli metni alıcıya gönderirken şifreli metinle birlikte gizli anahtarı da alıcıya güvenli bir şekilde göndermek gerekmektedir. Simetrik şifreleme algoritmaları çok hızlı bir şekilde şifreleme ve şifre çözme işlemlerini gerçekleştirebilmektedir [1, 2]. Tablo 1'de çeşitli simetrik şifreleme algoritmaları hakkında bilgiler verilmiştir [1].

Tablo 1 Bazı simetrik şifreleme algoritmaları bilgileri[1].

Algoritmanın Adı	Geliştiren	Tarihi	Tipi (Blok Uzunluğu)	Anahtar Uzunluğu	Döngü Sayısı	Çözülme Durumu	Kullanım Koşulları
DES (Data Encryption Standard)	IBM (ABD)	1977	Feistel Blok (64 bit)	56 bit (parity ile 64 bit)	16	Sağlam; 8 döngülü çeşidi çözülebiliyor; 16 zayıf anahtar	Serbest
IDEA (International Data Encryption Algorithm)	Lai-Massey, ETH Zurich (İsviçre)	1992	Blok (64 bit)	128 bit	8	Sağlam; 2^{51} zayıf anahtar	Ticari faaliyetler hariç serbest
RC2 (Rivest's Cipher veya Ron's Code2)	Rivest, RSA Data Security (ABD)	1992	Katar	2048 bite kadar	Bilinmiyor	Zayıflık bulunmadı	Algoritma RSA tarafından saklı tutuluyor
RC5 (Rivest's Cipher veya Ron's Code5)	Rivest, RSA Data Security (ABD)	1995	Blok (32, 64 veya 128 bit)	2048 bite kadar	255'e kadar	64 bit blok ve 12 döngü ile diferansiyel ve doğrusal şifre çözüme dayanıklı	Serbest
Blowfish	Bruce Schneier, Counterpane Systems (ABD)	1993	Feistel Blok (64 bit)	448 bite kadar	16	3 döngülü çeşidi diferansiyel şifre çözüme hassas	Ticari faaliyetler hariç serbest
FEAL (Fast Data Encipherment Algorithm)	Shimizu ve Miyaguchi (Japonya)	1988	Blok	FEAL-4 64 bit; FEAL-N 128 bit	FEAL-4 4 döngü; FEAL-N 31 döngü	Güvensiz; çeşitleri başarıyla çözümlendi	-
SAFER (Secure and Fast Encryption Routine)	Massey, Cylink Corporation (ABD)	1993	Blok (64 bit)	64 bit; 128 bit	10 döngüye kadar	İlk sürümlerinin anahtar açılımında zayıflıklar vardı	-
Skipjack (Clipper Chip)	NSA (ABD)	1993	Blok (64 bit)	80 bit	32	Algoritma gizli	Sadece özel entegre devre olarak bulunuyor
Lucifer	IBM (ABD)	1970?	Feistel Blok (64 bit)	128 bit	16	DES'in prototipi olduğundan zayıflıklar içermesi olasıdır	-
GOST 28147-89	I.A.Zabotin, G.P.Glazkov, V.B.Isaeva (Sovyetler Birliği)	1989	Feistel Blok (64 bit)	256 bit; 512 bit tanımlanabilir sübsitüsyon; 610 bit etken gizli bilgi	32	SSCB tarafından bütün gizlilik derecelerindeki bilgiler için uygun görülmüştür	Serbest
ASEKAL-21	Aselsan (Türkiye)	-	Doğrusal olmayan katar	57 bit ?	-	Ulusal olarak onaylanmış algoritma	Aselsan 2101, 2010 veri ve ses şifreleme birimlerinde kullanılıyor

2.2. Açık-anahtar (Asimetrik) şifreleme algoritmaları

Açık anahtarlı şifreleme algoritmaları simetrik şifreleme algoritmalarından radikal bir farklılık göstermektedir. Bu tip şifreleme algoritmaları açık (public) ve özel (private) anahtar olmak üzere iki ayrı anahtar kullanmaktadır.

Asimetrik algoritmalar da denilen açık anahtarlı algoritmalarda şifreleme için kullanılan anahtar ile şifre çözme için kullanılan anahtar birbirinden farklıdır. Anahtar çiftlerini üreten algoritmaların matematiksel özelliklerinden dolayı açık-özel anahtar çiftleri her kişi için farklıdır, diğer bir deyişle her kullanıcının açık-özel anahtar çifti yalnızca o kullanıcıya özeldir. Ayrıca şifre çözüm anahtarı (en azından makul bir zaman dilimi içerisinde) şifre anahtarından hesaplanamaz. Bu algoritmalara açık anahtarlı algoritmalar denmesinin sebebi şifre anahtarının halka (kamuya/genel kullanıma) açık olmasıdır. Bir yabancı bir iletiyi şifrelemek için şifreleme anahtarını kullanabilir, ancak sadece ilgili şifre çözüm anahtarına sahip bir kişi iletinin şifresini çözebilir. Bu sistemde, şifre anahtarına genellikle açık anahtar adı verilmektedir. Şifre çözüm anahtarı da genellikle özel anahtar olarak adlandırılmaktadır. Özel anahtar kimi zaman gizli anahtar olarak da adlandırılır, ancak simetrik algoritmalarla karışmaması için bu terim genelde kullanılmamaktadır.

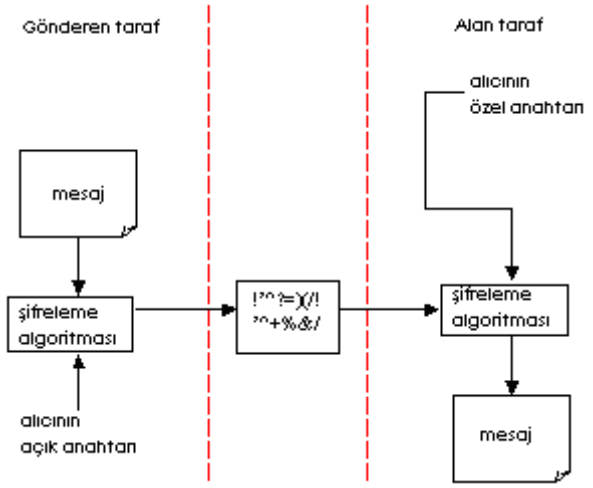
Bir kullanıcının açık anahtarıyla şifrelenen bir mesajı, yalnız ve ancak ona ait özel anahtar çözebilmektedir. Aynı şekilde, herhangi bir kullanıcının özel anahtarıyla attığı sayısal imzanın doğrulanabilmesi, yalnızca o kullanıcının açık anahtarını kullanarak mümkün olabilmektedir. Açık anahtar kamuya açıktır, elektronik kimlik

belgelerinin içinde diğer kişisel bilgilerle birlikte tutulur ve herkes birbirinin açık anahtarını e-kimliklerine ulaşmak suretiyle istediği zaman elde edebilir.

2.2.1 Şifreleme işlemi

Şifreleme açık ağlardan gönderilen bilginin başkaları tarafından görülmesinin (dinlenmesinin) istenmediği zaman yapılmaktadır. Bunun için çift anahtarlı bir şifreleme algoritması kullanılabilir. Buna göre, mesajı gönderen taraf, gönderilen bilginin sayısal içeriğini, mesajı alacak tarafın açık anahtarını, sayısal şifrelemede kullanmaktadır. Mesajı alan taraf da, şifreli mesajı çözmek için şifreli mesajın sayısal içeriği ve kendisinin özel anahtarına gereksinim duymaktadır [1, 3]. Şekil 2'de bu durum gösterilmiştir.

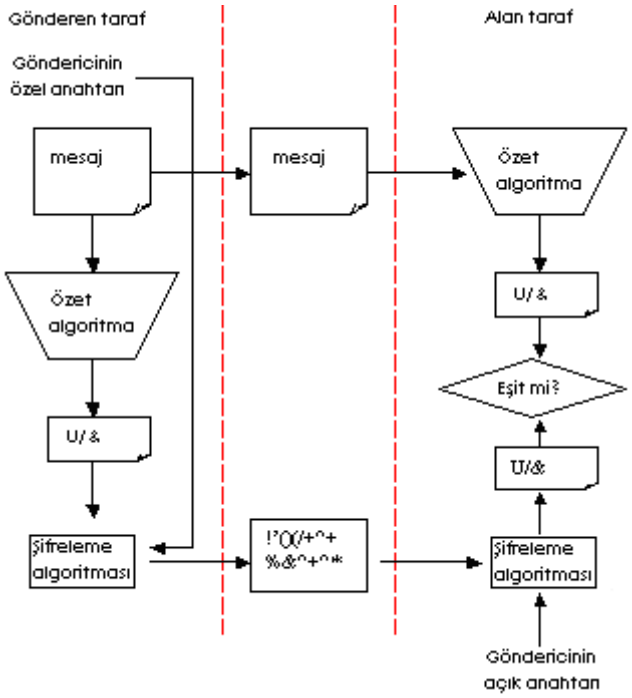
Burada dikkat edilecek olursa, şifreli mesajın üçüncü taraflar tarafından dinlenebilmesi ancak "özel anahtara" sahip olmaları ya da şifreli mesajı matematiksel yollarla deşifre etmeye çalışmaları ile mümkün olabilmektedir. "Güvenlik açısından iyi bir şifreleme" algoritması, özel anahtar olmadan şifreli mesajı deşifre etmeye imkân tanımayan bir algoritmadır.



Şekil 2 Şifreli mesaj gönderilmesi ve alınması

2.2.2. Sayısal imza

Sayısal imza elektronik mesaja eklenmiş bilgidir. Çift anahtarlı bir şifreleme algoritmasıyla hazırlanan sayısal imza, hem gönderilen bilginin sayısal içeriğinin değiştirilmediğinin hem de gönderen tarafın kimliğinin ispatlanması için kullanılır ve gönderilecek mesajdan üretilen "mesaj özetinin" sayısal içeriği, gönderen tarafın kendi özel anahtarına bağlı olarak oluşturulur. Sayısal imzanın doğruluğunu kanıtlamak için mesajı alan taraf, kendisine gelen mesajın ve sayısal imzanın sayısal içeriği ile gönderen tarafın açık anahtarını kullanmaktadır [7-9]. Şekil 3'de bu durum gösterilmiştir.



Şekil 3 Sayısal imzalı mesaj gönderilmesi ve alınması

"Mesaj özeti", gönderilecek mesajdan matematiksel yollarla üretilen sabit uzunlukta sayısal bilgidir. Bu işlem "hash" fonksiyonu olarak bilinir. Hash fonksiyonu bir kaç özelliği sağlar:

- Mesaj özeti anlamsız bir bilgidir.
- Hash fonksiyonu geri dönüşümü olmayan bir fonksiyondur. Diğer bir deyişle, herhangi bir mesajın özetine bakarak mesajın kendisini elde etmek mümkün değildir.
- Aynı özeti veren herhangi iki farklı mesaj bulmak da mümkün değildir.

Böylelikle, her mesajın farklı bir özeti olması ve dolayısıyla mesajda yapılacak en ufak bir değişikliğin imzayı geçersiz kılması sağlanmış olur. Sayısal imzalamada son adım, mesaj özetinin gönderen tarafın özel anahtarıyla şifrelenmesidir. Sayısal imza mesaja eklenir ve mesaj ile birlikte alıcıya gönderilir. Alıcının imzanın geçerliliğini kontrol etmesi iki adımda gerçekleşmektedir. Alıcı sayısal imzayı karşı tarafın açık

anahtarı ile çözerek varsayılan mesaj özetini elde eder. Diğer yanda mesajın tekrar özetini çıkarır. Son olarak bu iki özeti karşılaştırır. Bu özetlerin tıpatıp aynı olması, imzanın doğruluğunu gösterir.

Açık-anahtar şifreleme için pek çok algoritma bulunmaktadır. En yaygın olan iki tanesi RSA (Ron Rivest, Adi Shamir, Leonard Adleman) algoritması ve DSA'dır (Digital Signature Algorithm - Dijital İmza Algortiması). RSA, pek çok uygulamada kullanılan bir algoritmadır. Mesajları şifrelemek için kullanılabileceği gibi dijital imzalarda da kullanılabilmektedir. DSA, sadece dijital imza kullanımı içindir. Mesajları şifrelemek için kullanılmamaktadır [1, 10, 11].

2.3. Şifrelemede kullanılan anahtar boyutları

40 bitlik bir anahtar için $n=2^{40}$ veya n=1 099 511 627 776 (bir trilyon doksan dokuz milyar beş yüz on bir milyon altı yüz yirmi yedi bin yedi yüz yetmiş altı) olası anahtar söz konusudur. 1995'de yapılan bir yarışmada RC4 algoritması ile 40 bitlik bir anahtarla şifrelenmiş internet üzerinden yapılan bir kredi kartı işlemi, elinde sadece mütevazi bir bilgisayar laboratuarı olan bir öğrenci tarafından 3 buçuk saatte çözülmüştür [1].

Anahtarın deneme-yanılma yöntemiyle bulunmasını engellemek için, bugünkü süper bilgisayarlardan milyonlarca kat daha hızlı çalışan bir bilgisayarla bile milyarlarca yıl sürmesi için, kullanılan anahtarların uzunluğunun mümkün olduğunca büyük olması gerekmektedir.

Tablo 2'de farklı anahtar boyları için, saniyede bir milyon, bir milyar ve bir trilyon şifre deneyebilen bilgisayarlar için anahtar çözme süreleri verilmiştir. Tablo 3'te ise asimetrik şifreleme algoritması olan RSA şifreleme algoritması için kullanılan anahtar çiftlerinin farklı boyutlardaki oluşma süreleri ve şifreleme süreleri verilmiştir [1].

Tablo 2 Farklı anahtar boyutları için anahtar çözme süreleri

Anahtar Uzunluğu (n)	**Olası değer sayısı (2^n)**	**10^6 şifre/s hızında ortalama çözme süresi**	**10^9 şifre/s hızında ortalama çözme süresi**	**10^{12} şifre/s hızında ortalama çözme süresi**
32 bit	~$4x10^9$	36 dak	2.16 s	2.16 ms
40 bit	~10^{12}	6 gün	9 dak	1 s
56 bit	~$7.2x10^{16}$	1142 yıl	1 yıl 2 ay	10 saat
64 bit	$1.8x10^{19}$	292 000 yıl	292 yıl	3.5 ay
128 bit	$1.7x10^{38}$	$5.4x10^{24}$ yıl	$5.4x10^{21}$ yıl	$5.4x10^{18}$ yıl

Tablo 3 RSA algoritmasında farklı bit uzunluklarında anahtar oluşturma ve şifreleme süreleri

Bit sayısı	**Anahtar oluşturma süresi (sn)**	**Şifreleme (sn)**
64	*0.021*	*0.011*
128	*0.026*	*0.013*
256	*0.083*	*0.015*
512	*0.307*	*0.018*
1024	*2.985*	*0.106*
2048	*50.432*	*0.766*
4096	*798.625*	*18.687*

2.4. Simetrik ve asimetrik şifreleme algoritmalarının genel özellikleri

Simetrik ve asimetrik şifreleme algoritmalarının bazı önemli özellikleri tablo 4'te özetlenmiştir. Tablo 5'te ise iki algoritmanın özellikleri karşılaştırılmıştır. İki algoritmayı birbirinden ayırmak için, simetrik şifrelemede kullanılan anahtarı gizli anahtar (secret key) olarak, asimetrik şifrelemede kullanılan anahtarları ise, genel anahtar (public key) ve özel anahtar (private key) olarak adlandırılmaktadır. Özel anahtar daima gizli tutulur fakat simetrik şifrelemede kullanılan anahtarla karıştırılmaması için gizli anahtar'dan ziyade özel anahtar olarak adlandırılır.

Tablo 4 Simetrik ve asimetrik şifreleme algoritmalarının genel özellikleri[1]

Simetrik şifreleme algoritmaları	**Asimetrik şifreleme algoritmaları**
Aynı algoritma ve aynı şifreleme anahtarı hem şifreleme hem de şifre çözmede kullanılır.	*Şifreleme ve şifre çözmek için bir algoritma fakat şifreleme ve şifre çözme için farklı anahtarlar kullanılır*
Gönderici ve alıcı aynı algoritmayı ve aynı anahtarı kullanır.	*Gönderici alıcının açık anahtarını bilmelidir. Gönderici ile alıcının anahtar çiftleri birbirinden farklıdır.*
Şifreleme için kullanılan algoritma gizli tutulmalı	*İki anahtardan biri gizli tutulmalı diğeri erişime açık olmalıdır.*
Algoritma bilgisi ve şifreli metin örnekleri anahtarı belirlemede yeterli olmamalı	*Algoritma bilgisi, anahtarlardan birinin ve şifreli metin örnekleri,diğer anahtarı belirlemede yeterli olmamalı*

Tablo 5 simetrik ve asimetrik şifreleme algoritmalarının özelliklerinin karşılaştırılması

Özellik	**Simetrik şifreleme algoritmaları**	**Asimetrik şifreleme algoritmaları**
Gizlilik	*Sağlamaktadır*	*Sağlamaktadır*
Bütünlük	-	*Sağlamaktadır*
Kimlik doğrulama	-	*Sağlamaktadır*

İnkar edilememezlik	*-*	*Sağlamaktadır*
Performans	*Hızlı*	*Yavaş*
Güvenlik	*Anahtar uzunluğuna bağlı*	*Anahtar uzunluğuna bağlı*

3. Sonuç

Bu çalışmada simetrik ve asimetrik şifreleme algoritmaları incelenmiştir. Simetrik şifreleme algoritmaları tek bir anahtar kullanarak şifreleme ve şifre çözme işlemlerini geçekleştirmektedir. Bu algoritmalarda metin şifrelendikten sonra alıcıya bu şifreli metin gönderilirken, alıcıya gizli anahtarın da güvenli bir şekilde iletilmesi gerekmektedir. Bu durum simetrik şifreleme algoritmalarının en büyük dezavantajıdır. Asimetrik şifreleme algoritmalarında bu problem söz konusu değildir. Asimetrik şifreleme algoritmaları sayesinde alıcı ve gönderici taraflar kendilerine ait gizli anahtar oluşturabilirler ve verilerini bu anahtarla şifreleyebilirler. Asimetrik şifreleme algoritmaları çözülmesi zor matematiksel hesaplamalar üzerine kurulmuş algoritmalardır.

Asimetrik şifreleme algoritmalarının da dezavantajları bulunmaktadır. Bu algoritmaların güvenliğini sağlayabilmek için çok büyük asal sayılar kullanılmaktadır. Bu da zaman açısından çok büyük problemleri beraberinde getirmektedir.Asimetrik şifreleme algoritmalarını kullanan sistemler simetrik şifreleme algoritmalarını kullanan sistemlere göre çok daha yavaştır. Ayrıca asimetrik şifreleme algoritmalarının çok büyük sayılar kullanmasından dolayı donanımsal yapılara uyum sağlaması çok zor olmaktadır. Bundan dolayı sistemlerin hem simetrik hem de asimetrik şifreleme algoritmalarını birlikte kullanarak, simetrik şifreleme algoritmalarının dezavantajı olan

gizli anahtar güvenliği problemini ve asimetrik şifreleme algoritmalarının hız problemini ortadan kaldırabilmektedir.

Bir şifreleme algoritmasının performansı şu kriterlere göre belirlenebilir:

- Sistemin kırılabilme süresinin uzunluğu,
- Şifreleme ve çözme işlemlerine harcanan süre,
- Şifreleme ve çözme işleminde ihtiyaç duyulan bellek miktarı,
- Algoritmanın kurulacak sisteme uygunluğu.

Şifreleme bilimi hızla gelişen bir bilim dalıdır. Eski algoritmaların dezavantajlarını ortadan kaldıracak yeni şifreleme algoritmaları geliştirilmektedir. Sonuç olarak, asimetrik şifreleme algoritmalarında ki hızlı gelişim sayesinde dezavantajları ortadan kaldırabilirse günümüz teknolojisinde simetrik şifreleme algoritmalarının yerini alacağını göstermektedir.

Kaynaklar

[1] Kodaz H. Veri İletiminde Güvenlik İçin Şifreleme, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 2002.

[2] Yerlikaya T, Buluş E, Buluş N. Asimetrik Şifreleme Algoritmalarında Anahtar Değişim Sistemleri, Akademik Bilişim 2006 (Ab2006), 9-11 Şubat 2006, Denizli.

[3] Krishnamurthy M, Seagren ES, Alder R, Bayles AW, Burke J, Carter S, Faskha E. Basics of Cryptography and Enryption, How to Cheat at Securing Linux, 2008, 249-270.

[4] Stapko T. Security Protocols and Algorithms, Practical Embedded Security, 2008, 49-66.

[5] Kapor B, Pandya P. Data Encryption, Computer and Information Security Handbook, 2009, 395-421.

[6] Bellovin S.M. Cryptography and the Internet, *In Proceedings of CRYPTO '98*, August 1998.

[7] Aslan G.B. Sayısal İmza Sistemlerinin İncelenmesi, İTÜ, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 1999.

[8] Herranz J. Identity-based ring signatures from RSA, Theoretical Computer Science, Volume 389, Issues 1-2, 10 December 2007, 100-117.

[9] Ham L, Ren J. Efficient identity-based RSA multisignatures, Computer & Security, Volume 27, Issues 1-2, March 2008, 12-15.

[10] Saka Y. Bilgisayar Ağ Güvenliği ve Şifreleme, Muğla Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 2000.

[11] Shao Z. Batch verifying multiple DSA-type digital signatures, Computer Networks, 2001, Volume 37, Issues 3-4, 383-389.