RSA ALGORİTMASININ İYİLEŞTİRİLMESİ İÇİN YENİ BİR YAKLAŞIM

Arda AKSUOĞLU

Yüksek Lisans Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Eylül-2010

JÜRİ VE ENSTİTÜ ONAYI

Arda AKSUOĞLU'nun **"RSA Algoritmasının İyileştirilmesi için Yeni Bir Yaklaşım"** başlıklı **Bilgisayar Mühendisliği** Anabilim Dalındaki, Yüksek Lisans Tezi 27/07/2010 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı-Soyadı	**İmza**
Üye (Tez Danışmanı):	**Doç. Dr. YUSUF OYSAL**	**……...**
Üye	**: Yard. Doç. Dr. HÜSEYİN POLAT**	**……**
Üye	**: Yard. Doç. Dr. AHMET YAZICI**	**……**

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..... tarih ve ..... sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

RSA Algoritmasının İyileştirilmesi
İçin
Yeni Bir Yaklaşım

Arda AKSUOĞLU

Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Yusuf OYSAL
2010, 63 sayfa

Bilgi güvenliğinin giderek önem kazanmasından yola çıkarak bu tezde kriptoloji bilimi ele alınmıştır. Önemi, geçmişi, geleceği basitleştirilerek göz önüne serildikten sonra kriptografinin; asimetrik şifrelemenin, en önemli sistemi RSA incelenmiştir. RSA sistemi, modern şifreleme sistemleri için çok önemli olmasına rağmen tüm asimetrik sistemler gibi yavaş olması en büyük dezavantajıdır. Bu çalışmada da RSA sisteminin anahtar havuzunu genişleten fakat şifre açma (decryption) kısmını zaman bakımından yavaşlatan "Verimli RSA Şifreleme Şemasını" hızlandırmak için bazı şifre açma algoritmaları zaman karmaşıklıklarıyla (Büyük-O) beraber incelenmiş ve yeni bir algoritma önerilmiştir. Önerilen bu RSA algoritmasının nesne tabanlı programlama ile uygulaması yapılmış ve sonuçları birkaç RSA algoritması ile karşılaştırılmıştır. Önerilen sistemin avantajları, güvenliği, aksaklıkları, prensiplere uygunluğu irdelenmiştir.

Anahtar Kelimeler: Kriptografi, RSA, Şifreleme, Şifre Açma, Büyük-O analiz

ABSTRACT

Master of Science Thesis

A New Approach
for
Improving the RSA Algorithm

Arda AKSUOĞLU

Anadolu University
Graduate School of Science
Computer Engineering Program

Supervisor: Assoc. Prof. Yusuf OYSAL
2010, 63 pages

Taking the importance of information safety into account in today's world, this paper examines cryptology. After simplifying the importance, history and future of cryptography, the most important system of asymmetric cryptography, RSA, has been studied throughout the paper. Although RSA is of great importance for current encoding systems, its working slowly, like all the other asymmetric systems, is one of its greatest drawbacks. In this study, some algorithms have been investigated, together with their time complexity, in order to fasten "Efficient RSA Encryption Scheme" which not only contributes to RSA but also retards decryption. And, as a result a new algorithm has been suggested. This suggested algorithm has been practiced by object-oriented programming and its results have been compared to those of a few RSA algorithms. Therefore, this paper examines the advantages, safety, pitfalls of the suggested system, as well as, this system's being suitable for the principles or not.

Key Words: Cryptography, RSA, Encryption, Decryption, Big-O analysis

TEŞEKKÜR

Bu tez çalışması sürecinde destek ve yardımlarını esirgemeyen danışman hocam Sayın Doç. Dr. Yusuf Oysal'a ve hocam Sayın Yard. Doç. Dr. Hüseyin Polat'a; her türlü yardımlarından dolayı arkadaşlarım Sevcan Yılmaz, Eyüp Işık ve Korhan Turhan'a ve de bilim insanını her türlü destekleyen Tübitak-BİDEB'e teşekkürü borç bilirim.

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

SİMGELER ve KISALTMALAR DİZİNİ

E	: Şifreleme (encryption)
D	: Şifre açma (decryption)
M	: Açık Mesaj
C	: Şifreli metin
e	: RSA'da açık anahtar
d	: RSA'da özel anahtar
r	: İki asal sayı (*p ve q*) arasındaki fark
KU	: Açık anahtarlar
KR	: Özel anahtarlar
$\Phi(n)$	: *n* ile aralarında asal olan tam sayıların sayısı
OBEB	: Ortak bölenlerin en büyüğü

Alt indisler

KU_B	: *B* kullanıcısının açık anahtarının kullanılması
KR_B	: *B* kullanıcısının özel anahtarının kullanılması

1. GİRİŞ

Bilgilerin paylaşıldığı, yani iletişimin olduğu andan itibaren bilgi gizliliği ihtiyaç halini almıştır. Bu ihtiyaç "Kriptoloji" ve "Steganografi" bilimini doğurmuştur. Bu bilimlerden Steganografi, bilginin varlığını koruyarak bilgiyi gizlemeyi konu edinir. Örneğin bilginin bir resmin içine çıplak gözle bakıldığında görülemeyecek şekilde gizlenmesi Steganografi'nin ilgi alanıdır. Diğer bilim dalı Kriptoloji ise bilginin içeriğini gizlemeyi konu edinir. İçeriğin gizlenerek iletilmesi, mesajın başkalarının eline geçmesi halinde bile anlaşılamamasını sağlar. Kriptoloji bu temel üzerine kurulmuştur.

Kriptoloji ilk olarak devletlerin ya da orduların kullandıkları gereç olsa da günümüzde herkesin hayatına girmiştir. 1990'lı yıllardan başlayarak yaşanan hızlı teknolojik gelişmeler; bilgisayarların ve internetin modern hayatın her alanına girmesi ve vazgeçilmez bir biçimde kullanılması kriptoloji biliminin önemini giderek arttırmıştır. Özellikle kimlik kanıtlanması ve kimlik bilgilerinin gizlenmesi gibi konularda güvenle kullanılabiliyor olması kriptografik yöntemleri hayatın değişilmezleri arasına sokmuştur. Örnek vermek gerekirse; elektronik bankacılık, elektronik ticaret, para transferleri, kamu hizmetleri, ücretli televizyon yayıncılığı gibi birçok alan kriptografik yöntemlerin gündelik hayatta kullanıldığı alanlardır.

Kriptolojiyi öğrenmek ve onun modern toplumlara sağladığı avantajlardan yararlanmak hem kişisel gizliliğimiz hem de elektronik dünyadaki güvenliğimiz için kaçınılmazdır. Bu tezde de kriptoloji bilimi içerisinde önemli yer tutan asimetrik şifreleme algoritmalası RSA incelenmiştir.

1.1. Kriptoloji

Kriptoloji; Yunanca, "kryptos"(gizli) ve "logos" (bilim) kelimelerinin birleşimden gelmektedir. Basit anlamda şifreli belgeler, gizli yazılar bilimidir; "Kriptografi" ve "Kripto analiz" diye iki ana dala ayrılmıştır.

Kriptografi; Yunanca, "kryptos" ve "graphein" (yazmak) kelimelerinin birleşiminden doğmuştur. Belgelerin şifrelenmesi ve şifresinin çözülmesi için

kullanılan yöntemlere verilen addır. Şifreleme ve şifre açmakta kullanılan tekniklerin tümünü inceleyen bilim dalıdır.

Kripto analiz; bir şifreleme sistemini veya sadece şifreli mesajı inceleyerek, şifreli mesajın açık halini elde etmeye çalışan kriptoloji disiplinidir. Kaba tabiriyle, kriptografi bilimi ile şifrelenen metinler, kripto analiz bilimi ile kırılmaya çalışılır.

Kriptoloji, bilgi gizliliğinin önem kazandığı yani insanoğlunun iletişime gereksinim duyduğu andan itibaren hep ön planda olmuştur. Binlerce yıl önce; devletler, imparatorluklar gizli ve önemli bilgileri düşmanın eline geçmeden iletebilmek için özel yetiştirilmiş ulaklar ve güvercinler kullanmışlardır. Fakat bu; mesajın başkalarının eline geçmesine engel olamamıştır. İşte bu devrede mesajın anlamını da gizleme gereği; kriptolojiyi ortaya çıkarmıştır.

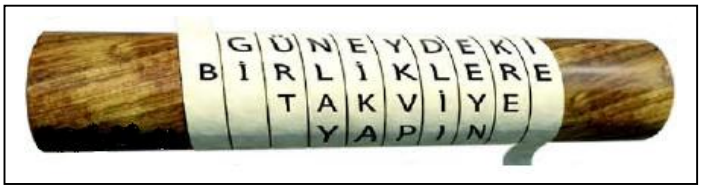
Binlerce yıl önce, bilgileri kodlama ile başlayan kriptoloji; yani sözcüklerin veya cümlenin başka bir sözcük, sayı ya da sembol ile yerini değiştirerek göndermek, o dönem için mesajın başkaları tarafından anlaşılmasını engellemek için yeterli olmuştur. Fakat gelişen teknoloji ile birlikte bu yöntemlerde hızla tarih sayfalarında yerlerini almıştır. Artık kriptoloji biliminde; matematik temellerine, bilgisayarların işlem güçlerine dayalı sistemler hâkimiyet sürmektedir.

1.1.1. Kriptolojinin tarihi

M.Ö. 2000'li yıllarda Nil nehri kıyısında küçük bir şehir olan Menet Khufu'daki bir kâtibin, efendisinin hayatını anlatırken kullandığı hiyeroglifler kriptoloji tarihinin ilk kayıtları olarak ele alınır. Bu hiyerogliflerin şifreleri çözüldüğünde, daha önce hiç kullanılmamış olan simgelerin kullanılmış olduğu görülmüştür. Bu simgelerin, sadece ilgili kişi tarafından anlaşılabilir olduğu ve bu kişi dışındakiler tarafından anlaşılmaması için kodlandığı sonucuna varılmıştır. Bunun da bilgi güvenliği sağlamak üzere uygulanmış bir kripto tekniği olduğu kabul edilmektedir.

Askeri haberleşmede, kriptografi kullanan ilk ulus Ispartalılardır. Ispartalılar'ın M.Ö. 5.yüzyılda geliştirdikleri Şekil 1.1'deki cihaz tarihin ilk yer değiştirme sistemi olarak askeriye tarafından kullanılıyordu. Bu cihaz belli

kalınlıkta bir tahta silindirden ve silindirin etrafına eğik biçimde sarılmış papirüs ya da ince, deri bir şeritten oluşuyordu. Gizli mesaj silindir boyunca silindire sarılı şerit üzerine yazılıyor, daha sonra şerit silindirden çözülüyordu. Birbirinden ayrılan harfler yeniden aynı kalınlıkta bir tahta silindire sarılmadıkça hiçbir anlam ifade etmiyordu.

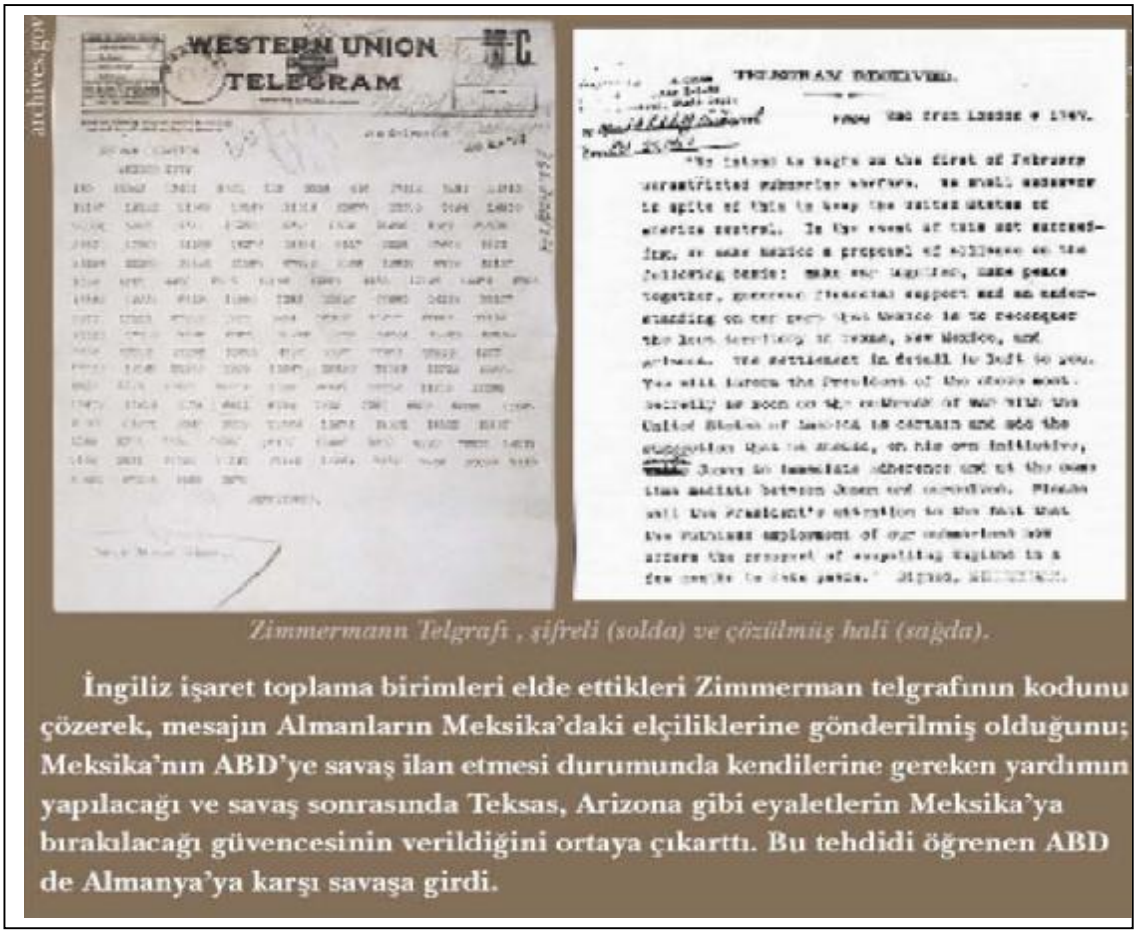


Şekil 1.1. Ispartalılar'ın Kripto Cihazı

İlk ciddi kripto analiz çalışmaları ise Araplar tarafından yapıldı. O döneme kadar kripto kırma faaliyetleri yok denecek kadar azdı. Araplar kriptoloji çalışmalarına edebiyatta ve matematikte çağın ilerisinde oldukları M.S. 600′lü yıllarda başladılar. Şifre anlamına gelen İngilizce "cipher" ve Fransızca "chiffre" sözcükleri bu dillere Arapçadan (cifr ya da cifir) geçmiştir. Arapların kriptoloji konusunda yazdıkları ilk eser, Abdurrahman el-Halil tarafından M.S.718 yılında kaleme alınan "Kitab-ül Muamma" adlı kitaptır. Bu kitapta Abdurrahman el-Halil, Bizans imparatoru tarafından gönderilen Yunanca şifreli mektubun çözümünü verir [1].

Günümüz kriptoloji sistemlerinin temeli ise dünya savaşları ile atılmıştır: Birinci Dünya Savaşı sırasında kriptografinin çok yoğun kullanımı ve savaşın haberleşme teknolojisinin ilerlemesine katkısı, savaş sonrasında kriptolojinin gelişen teknolojiden daha fazla yararlanmasına neden olmuştur. Birinci Dünya Savaşı yıllarında telsiz haberleşmenin icadı ile kriptolojinin önemi çok artmıştır. Telsiz haberleşmenin doğası gereği, iletilen mesajların sadece gönderildiği kişi tarafından değil, radyo dalgalarını alabilen herkes tarafından dinlenebilmesi söz konusudur. Bu nedenle telsiz haberleşmesinde bilgi güvenliğini sağlamak üzere yeni teknikler geliştirilmesi zorunlu hale gelmiştir. Birinci Dünya Savaşı'nda Almanlar "ADFGVX" olarak adlandırdıkları sistemleri ve "Kod Kitabı" yöntemini kullanmışlardır. Bu yöntemle şifrelenecek metindeki her kelimeye karşı bir sayı grubu kullanılmaktaydı. Örneğin Zimmermann telgrafı olarak bilinen mesajda "Februar" kelimesi "13605, "fest" kelimesi 13732 sayıları ile kodlanıyordu.

Birinci Dünya Savaşı sırasında kripto analiz teknikleri de oldukça gelişmişti. Nitekim İngilizler, Almanların şifreleme sistemini çözerek savaşa yön vermişlerdir (Şekil 1.2). Bu nedenle daha güçlü şifreleme sistemlerine gereksinim doğdu. Sonuçta daha yeni kriptografik yöntemler ortaya çıkmış ve bu yöntemle çalışan cihazlar bir sonraki dünya savaşında gizli haberleşmeye yön vermiştir.



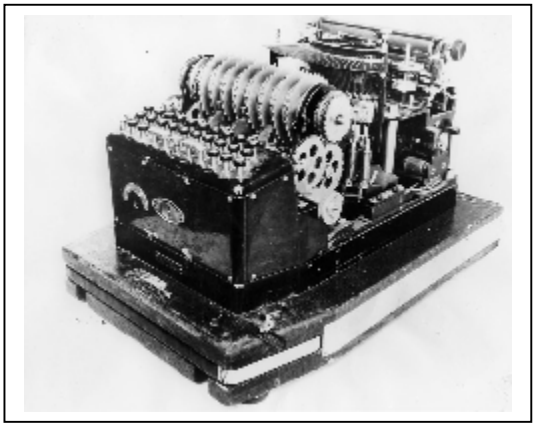
Zimmermann Telgrafı , şifreli (solda) ve çözülmüş hali (sağda).

İngiliz işaret toplama birimleri elde ettikleri Zimmerman telgrafının kodunu çözerek, mesajın Almanların Meksika'daki elçiliklerine gönderilmiş olduğunu; Meksika'nın ABD'ye savaş ilan etmesi durumunda kendilerine gereken yardımın yapılacağı ve savaş sonrasında Teksas, Arizona gibi eyaletlerin Meksika'ya bırakılacağı güvencesinin verildiğini ortaya çıkarttı. Bu tehdidi öğrenen ABD de Almanya'ya karşı savaşa girdi.

Şekil 1.2. Zimmermann telgrafı [2]

İkinci dünya savaşında kriptolojinin önemi çok daha iyi anlaşılmıştır. Kriptoloji sayesinde devletler, düşmanlarının eline geçmeden istedikleri birimleriyle haberleşebiliyorlardı. Bunun en güzel örneklerinden biri de, Almanların kullandığı ve savaşa yön verdiği şifreleme makinesi olan Enigma'dır. (Şekil 1.3a) Enigma mekanik ayarla yönlendirilen elektronik bir cihazdı. 1918 yılında Arthur Scherbius tarafından ticari uygulamalar üzere geliştirilmiş ve ilk olarak bankalar arası haberleşmede kullanılmıştı. Alman ordusunun Enigma'yı

kullanma kararı sonrasında İkinci Dünya Savaşı süresince toplam yüzbin kadar Enigma kripto cihazı üretilmiştir. Enigma kripto cihazı 21. yüzyıla kadar en yüksek miktarda üretilen kripto cihazı olma özelliğini korumuştur. Almanların savaş yıllarının başlarında kazandıkları müthiş başarının nedenlerinin başında düşmanlarının anlamadığı ve çözemediği bir şifreleme tekniği kullanmalarıydı. Alman ordusu, muhabere birimindeki görevlilere gönderilecek olan mesajı Enigma cihazının tuş takımı üzerinden yazıyor, mesaj cihaz içersinde şifreleniyor ve elde edilen şifreli mesaj telsiz operatörü kullanılarak, radyo dalgaları ile karşı tarafa gönderiliyordu. (Şekil 1.3b)



(a)

Enigma, Alman General Panzer'in muhabere birliği tarafından kullanılıyor.

(b)

Şekil 1.3. İkinci Dünya savaşında (a) Enigma ve (b) kullanılması [2]

Savaşın ilerleyen yıllarında İngiliz ve Polonyalı matematikçilerin uzun uğraşları sonucunda yeni kripto analiz yöntemleri geliştirmeleri ile Enigma'nın güvensizliği ortaya çıkmıştır. Böylece Almanların başarısı bitmeye yüz tutmuştur. O dönemde bulunan kripto analiz yöntemlerini uygulayabilmek için; kısa sürede çok sayıda işlem yapan cihazlara ihtiyaç doğdu. Bu da bilgisayarların ilkel versiyonlarının bulunmasını sağladı. Böylelikle kriptoloji biliminde yeni bir dönem başladı.

1.1.2. Kriptolojinin geleceği

Tarih boyunca kriptograflar ve kripto analistler sürekli bir yarış içinde olmuşlardır. Kriptografi biliminde, asimetrik kriptografinin bulunuşuyla kriptograflar bir adım öne geçmiştir ve günümüzde kriptografi bilimi altın çağını yaşamaktadır. Mevcut bilgisayarların işlem gücü, matematiksel formüllere dayanan günümüz şifreleme sistemlerini kırmak için yeterli olmamaktadır. Bilgisayarların bir saniyede yaptığı işlem sayısı her iki senede yaklaşık olarak iki katına çıkmasına rağmen bu artış günümüz sistemleri kırmak için yeterli olmamaktadır.

Kripto analistlerin bugünün bilgisayarlarından trilyonlarca kat daha fazla işlem yapabilen bilgisayarlara ihtiyacı vardır. Bu ihtiyaç dikkatlerin kuantum bilgisayarlara çevrilmesine sebep olmuştur. Fakat kuantum bilgisayarlar şu an için sadece teoridedir. Günümüzdeki elektroniğin özeti olan 1/0 mantığı yerini atomik süperpozisyonlara dayalı hem 1 hem 0 olabilen sistemlere bırakırsa, veri depolama kapasitesi ve işlem hızı açısından akıl almaz bir artış olacaktır ve şu an hayal gibi gelen yüksek hızda uygulamalara olanak verecektir. Kuantum bilgisayarların getireceği hız bir örnek ile anlatılırsa: Bir anakentte Leyla'sını arayan Mecnun'u ele alalım. Fakat Mecnun; Leyla'nın adresini bilmemektedir. Bu durumda Mecnun şehirdeki bütün kapıları çalarak Leyla'nın o evde oturup oturmadığını sormak zorundadır. Bu şekilde Mecnun'un Leyla'yı bulması aylarını, yıllarını alır. Bu durumun günümüz bilgisayarları yansıttığını düşünüp, kuantum bilgisayarların davranışına geçelim. Eğer Mecnun kuantum bilgisayar gibi davranırsa, Leyla'yı bulması saniyesini alır. Mecnun kendisini klonlayarak bir anda şehirdeki bütün kapıların önünde belirir. Doğru kapıyı bulduğunda diğer kapılardaki bütün Mecnunlar kendini imha eder. Teknolojinin gelişim hızı dikkate alındığında kuantum bilgisayarların 30 yıl sonra piyasada olabileceği ön görülmektedir.

Günümüz kripto analistlerin asıl hedefi modern kriptografinin temel taşı olan RSA'yı kırabilmektir. RSA en önemli askeri, diplomatik, ticari ve suç örgütlerinin iletişimlerini korumakta kullanılmaktadır [3]. Aslında RSA uygulaması basit bir çarpanlara ayırma problemidir, fakat çok büyük sayılar kullanıldığı için çözümü basit değildir. Matematikçiler bu konuyu yüzyıllardır

incelemelerine rağmen eski yöntemlerden ileri gidememişlerdir. Belki de matematik yasalarında hiçbir zaman çarpanlara ayırma işini yapan bir yöntem bulunamayacak; bu yüzden işlem gücü yüksek bilgisayarlar geliştirilip klasik çarpanlara ayırma yöntemi ile her bir asal sayının tek tek denenip N sayısına bölünüp bölünmediği kontrol edilmelidir. Kuantum bilgisayarlar saniyede katrilyonlarca işlem yapabileceği için çarpanlara ayırma işi kısa sürede çözülecek ve RSA sisteminin süresi dolacaktır.

Kuantum bilgisayarlar ile günümüz şifreleme sistemlerinin süresi dolacağı gibi kırılması imkânsız şifre sistemleri de doğacaktır. Bu sistemler de *Kuantum Kriptografi* sistemleridir. Kuantum kriptografi aynı zamanda mesajın şifrelenmesinde kullanılacak anahtarın güvenli bir biçimde alıcı ve verici arasında değişimini de öngörür. Kuantum kriptografi ile anahtarın araya giren bir kişi tarafından bile kopyalanamayacağı bir sistem ortaya çıkacaktır. Kuantum kriptografisi temel bir fizik kanunu olan Heisenberg'in belirsizlik ilkesine dayanır. Bu ilkeye göre kuantum mekaniğinin temel öğesi olan bir fotonun aynı anda iki özelliği bilinememektedir. Bu da iletişim kanalındaki bir fotonun klonlanmasını imkânsız hale getirir. Kuantum kriptografisi bu özellikten faydalanarak güvenilir anahtar ve şifreleme iletimi sağlar. Bu da kuantum kriptografisinin geleceğin şifrelemesi olacağını gösterir. Ancak kuantum kriptografisinin ne zaman kullanılmaya başlanacağı belli değildir. Kuantum bilgisayar teorisi hayata geçerse şu anda kullanılan hiçbir sistem güvenli olamayacaktır. Buna ek olarak kuantum kriptografisi ile sonsuza dek güvenli haberleşme doğacağı varsayılmaktadır.

1.1.3. Ülkemizde kriptoloji çalışmaları

Ülkemizde yakın bir geçmişe kadar Türkiye Cumhuriyeti'nin ve Türk Silahlı Kuvvetleri'nin bilgi güvenliğini sağlamak için gerekli olan kriptografik cihazlar yurt dışından tedarik ediliyordu. Yurt dışından tedarik edilen cihazlar hem zaman hem de ekonomik olarak kayba sebep oluyordu. Ayrıca tedarik edilen cihazların güvenliğinden de şüphe ediliyordu. İhtiyaç duyulan cihazların ulusal kaynaklarca üretilmesi düşüncesi ilk olarak 1974 Kıbrıs Barış Harekâtı sonrası gerçekleşti. Gemel Kurmay Başkanlığı'nın TÜBİTAK Marmara Araştırma Merkezi'ni gemilerimizde kullanmak üzere "milli" bir kripto cihazı geliştirmek

üzere görevlendirmesi ile ülkemizdeki ilk çalışmalar başlamıştır. İlk meyvesini 1978 yılında vermiştir. Üretilen bu kripto cihazın adı MİLON-1 dir. Ülkemizdeki bu gelişimi bu cihazın yeni versiyonları, MİLSEC(Milli Ses Emniyet Cihazı) gibi farklı yapılar izlemiştir [4,5].

1972 yılında kurulan Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü (UEKAE) ülkemizde halen daha bu alanda hizmet vermektedir. Anahtar yönetim sistemleri, haberleşme ürünleri gibi birçok ürün ile bu alana katkıda bulunmaya devam etmektedir. UEKAE, 1994 yılında "Kripto Analiz Merkezi Teşkili Projesi" kapsamında kriptografik sistemlerin analiz ve tasarımının yapılması amacıyla gerekli alt yapıyı oluşturmuştur. Bu alanda da çalışmalarına devam etmektedirler.

UEKAE, bilgi güvenliği ve ileri elektronik teknolojileri alanlarında NATO'da da aktif rol oynayan bir kurumdur. AR-GE (Araştırma- Geliştirme) sonucunda üretilen kriptografik cihaz ve algoritmalar, NATO Askeri Komitesi tarafından, NATO ve NATO üye ülkelerin tüm gizlilik seviyelerindeki haberleşmesi ve bilgi güvenliğinin sağlanması için onaylanmış ve bugüne kadar birçok ürünün NATO ülkelerine satışı gerçekleştirilmiştir.

1.1.4. Kriptoloji terminolojisi

Bir sanat, bir bilim ya da bir teknik dalında özel olarak kullanılan terimlerin tümüne terminoloji denir. Kriptoloji alanında sürekli karşılaşılan terimler şu şekilde özetlenebilir.

Açık Metin (Plaintext): İçeriği gizlenecek verinin, bilginin ilk halidir. Üzerinde herhangi bir oynama yapılmamış olan mesajın kendisidir.

Şifreleme (Encryption): Verinin kriptolaşmasıdır. Şifreleme süreci mesajı gönderen tarafından gerçekleştirilir. Hedef, içeriğin gizli kalmasıdır.

Şifreli Metin (Ciphertext): Verinin standart ya da özel algoritmalar ile kriptolaşmış hali. İçeriği gizli olan mesaj.

Şifre Açma (Decryption): Şifreleme sürecinden geçmiş mesajı alan kişinin açık metne ulaşmasıdır.

Anahtar: Verinin gizlenmesi ve şifreli metnin çözümü için gerekli olan sayısal, görsel veya nesnel değerler.

Açık Anahtar (Puplic key): Herkesin görebileceği, kullanabileceği anahtar. Her kripto sistemde bulunmaz. Asimetrik sistemlerde bulunur. Örneğin Mecnun; Leyla'ya mesaj göndereceği zaman Leyla'nın açık anahtarını kullanarak mesajı şifreler.

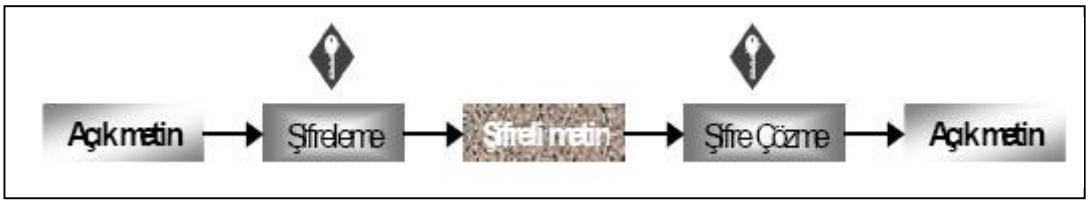
Özel anahtar (Private key): Asimetrik sistemlerde, kişinin sadece kendisine ait olan anahtarıdır. Bu anahtar başka kimse ile paylaşılmaz. Şifreli metni çözmek ve açık metne ulaşmak için kullanılır.

Gizli anahtar (Secret key): Simetrik sistemlerdeki anahtardır. İki kişinin kendi arasında anlaşarak belirlediği anahtardır. Bu anahtarı iki kişinin dışında kimse bilmez. Karışıklığı önlem adına simetrik sistemlerdeki anahtar *gizli*, asimetrik sistemlerdeki anahtar *özel* olarak adlandırılır.

Kripto Sistem: Şifreleme ve şifre çözme yapan bir sistemdir.

1.2. Kriptografi

Kriptografi, veriyi yalnızca okuması istenen şahısların okuyabileceği bir şekilde saklamak veya güvenli olmayan ortamlardan (örneğin internet veya yerel ağlar; network) iletilmesini sağlamak amacıyla kullanılan bir teknolojidir. (Şekil 1.4)Ya da başka bir deyişle kriptografi Mecnun'un; Leyla'ya gönderdiği bilginin güvenliğini sağlayan bilimdir. Bilgi güvenliği; başkası tarafından dinlenme, bilginin değiştirilmesi, kimlik taklidi gibi tehditlerin ortadan kaldırılması ile sağlanır. Bu güvenliğin sağlanması için veri, bilgi vb. matematiksel yöntemler kullanılarak kodlanır ve başkalarının okuyamayacağı hale getirilir. Bu matematiksel kodlamaya "*kripto algoritması*" adı verilir.



Şekil 1.4. Kriptografinin çalışma şekli

Kripto algoritmasının güçlü olması tehditlere karşı dirençli olması hiç şüphesiz ki en önemli özelliktir. Şifreleme veya şifre açma anahtarlarından biri üçüncü şahıslar tarafından ele geçirebilir ancak sistem güvenliği sağlayan tüm kısımların böyle bir durumda çözülememesi önemlidir.

"Bu dünyada kriptografinin iki türü vardır: *Kardeşinizin belgelerinizi okumasını engelleyen kriptografi, ve hükümetlerin belgelerinizi okumasını engelleyen kriptografi*." Bruce Scheier (Counterpane Internet Security şirketinin kurucusu)

Kriptografi, yukarıdan da anlaşıldığı gibi güçlü ya da zayıf olabilir. Kriptografinin gücü; şifreli metni, açık metne çevirmek için gerekli zamanın ve araçların kapasitesiyle ölçülür. Bunun yanında güçlü kripto algoritmaları her türlü tehditlere karşı güvenlik prensiplerini karşılamalıdır.

1.2.1. Kriptografi prensipleri

Kriptografi prensipleri zaman içinde, bilgisayar sistemlerine karşı ortaya çıkmış tehditler ve yaşanmış olaylar sonucunda ortaya konmuştur. Yani her bir hizmet, belli bir grup potansiyel tehdide karşı sistemi korumaya yöneliktir. İşte bu prensipler

- Gizlilik
- Kimlik Doğrulama
- Bütünlük
- İnkâr Edememe
- Güvenilirlik, diye beş ana başlık altında toplanabilir [6,7].

1.2.1.1. Gizlilik (*Confidentiality*)

Bilginin sadece erişim hakkı olan yetkili kişilerce erişilebilir olmasının temini olarak tanımlanabilir. Bir diğer tarif ile gizlilik, bilginin yetkisiz kişilerce açığa çıkarılmasının engellenmesidir. Basit bir örnek ile ifade edilirse; normal yoldan gönderilen bir mektubun, alıcı kişiye giderken yolda herhangi bir kişi tarafından okunmasını (örneğin postacı) engelleme amacı güder. Yazılan mektup açık metin seklindedir ve herhangi bir kişi tarafından zarfın açılması halinde, gönderilmiş mektup okunabilir. Şifreleme bu açık metnin, şifrelenerek yazılması

işlemini gerçekleştirir. Bu sayede mektubun yolda giderken herhangi bir kişi tarafından açılması halinde yazı açık metin olmadığı için mesajın okunması engellenecektir.

Gizlilik, fiziksel ortamlarda güvenlikten, matematiksel algoritmalara kadar varan birçok yaklaşımla sağlanır. Parola dosyalarının çalınması, ağ üzerindeki trafiğin gözetlenmesi ve kaydedilmesi, yetkili kullanıcının fark ettirilmeden gözetlenmesi ile kullanıcıya ait kullanıcı adı ve parola gibi özel bilgilerin alınması, sisteme giriş yapan kullanıcının bilgisayarını saldırganın izinsiz kullanması gibi durumlar Gizlilik prensibi kapsamında değerlendirilir.

1.2.1.2. Bütünlük (*Integrity*)

Bilgi veya mesaj bütünlüğü bilginin ağdaki yolculuğu sırasında değişikliğe uğramadığından emin olmak için gereklidir. Yani, veriyi göndericiden çıktığı haliyle alıcısına ulaştırmaktır. Bu durumda veri, haberleşme sırasında izlediği yollarda değiştirilmemiş, araya yeni veriler eklenmemiş, belli bir kısmı ya da tamamı tekrar edilmemiş ve sırası değiştirilmemiş şekilde alıcısına ulaşır.

Normal yoldan gönderilen mektup yine üçüncü kişiler tarafından yolda orijinal seklin dışında başka bir şekle dönüştürülerek yolculuğuna devam ettirilebilir. Yazılan mektup açık metin şeklindedir ve içerik okunabilmektedir. Okunabilen bu açık metin kötü niyetli kişiler tarafından yolda değiştirilerek, alıcıya farklı içerikle gönderilebilir. Bu prensibi bazı asimetrik şifreleme algoritmaları düz metin üzerinde işlem yaparak sayısal bir sonuç çıkararak sağlar. Çıkarılan sonuç gönderilen yazının üzerinde en ufak bir değişiklik yapıldığında, algoritma aynı olduğundan değişecektir.

1.2.1.3. Reddedilemezlik (*Non-Repudation*)

Alıcının veya göndericinin iletilen mesajı inkâr edememesidir. Bu hizmet sayesinde, ne gönderici alıcıya bir mesajı gönderdiğini ne de alıcı göndericiden bir mesajı aldığını inkâr edebilir. Bu prensip gönderici ve alıcı arasında ortaya çıkabilecek iletişim sorunları ve anlaşmazlıkları en aza indirmeyi amaçlar. Bu hizmet, özellikle gerçek zamanlı işlem gerektiren finansal sistemlerde kullanım alanı bulmaktadır.

1.2.1.4. Kimlik belirleme (*Authentication*)

Kimlik belirleme bilginin doğru kaynaktan alındığını doğrulamak için kullanılır. Kimlik belirleme, alıcının veya göndericinin yahut kullanıcının iddia ettiği kişi olduğundan emin olunmasıdır. En basit şekli ile bir bilgisayar sistemine giriş yaparken parola girilmesi de kimlik belirlemesidir.

Bu prensip iletimi gerçekleştirilen bir mesajın göndericisinin gerçekten gönderen kişi olduğu garanti eder. Kişiye ulasan bir mektubun üzerinde bulunan gönderici ismi her zaman doğru olmayabilir. Kötü niyetli kişiler tarafından gönderici isimleri farklı yazılarak kişilere mektup yollanabilir (spam mailler gibi). Şifreleme, bilim olarak bu mektuplar üzerine özel imzalar (*signature*) ekleyerek, mektubu gönderen kişinin gerçekten mektubu gönderen kişi olduğundan emin olmayı sağlayabilir. Özel algoritmalarla oluşturulan imzalar, alıcı kişi tarafından belirli yöntemlerle doğrulanabilir. Bu imza oluşturma ve doğrulama işlemi dijital imza olarak adlandırılır

1.2.1.5. Güvenilirlik (*Consistency*)

Sistemin öngörülen ve beklenen davranışı ile elde edilen sonuçlar arasındaki tutarlılık durumudur. Sistemin kendisinden beklenen şeyi eksiksiz ve fazlasız olarak her çalıştırıldığında tutarlı bir şekilde yapması olarak tanımlanabilir.

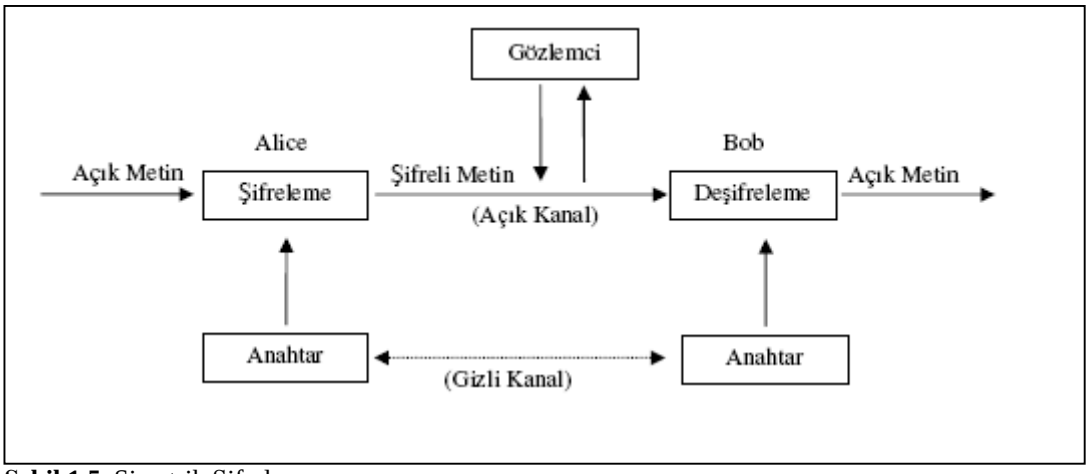
1.3.2. Kriptografi çeşitleri

Özellikle ikinci dünya savaşından sonra kriptoloji biliminde bilgisayar ve matematiğin etkin bir şekilde kullanılmaya başlanmasıyla şifreleme sistemleri 1970’lerde yeni bir boyut kazandı. Kripto sistemler simetrik ve asimetrik sistemler olarak ikiye ayrıldı.

1.3.2.1. Simetrik şifreleme

Geleneksel veya gizli anahtarlı şifreleme olarak da adlandırılan simetrik şifrelemede, şifreleme ve şifre açma için tek bir anahtar kullanılır. Gönderen taraf, mesajı bir anahtarla şifrelerken, alıcı taraf da aynı anahtarı kullanarak şifreyi açar. Alıcı ve göndericinin simetrik şifreleme kullanarak güvenli bir şekilde haberleşmesi için, bir anahtar üzerinde anlaşmaları ve bu anahtarı gizli tutmaları gerekmektedir. Eğer bu kişiler ayrı konumlarda bulunuyorsa, taşıyıcının, telefon sisteminin ya da diğer tasıma ortamlarının özel anahtarın saklanabilmesi açısından yeterli güvenilirlikte olması gerekmektedir. Çünkü anahtarı ele geçirecek her kişi, şifreyi çözebilir. Anahtarların üretimi, iletimi ve saklanması anahtar yönetimi olarak adlandırılır ve tüm şifreleme sistemleri anahtar yönetimi sorunlarıyla uğraşmak durumundadır. Anahtarların gizli kalmasını gerektirdiğinden dolayı, simetrik şifreleme, özel anahtar yönetiminde oldukça sıkıntı yasamaktadır.

DES, Blowfish, Twofish, AES, CAST128, RC5 bazı simetrik şifreleme algoritmalardır. Bu algoritmaların en büyük avantajı basit ve kolay uygulanabilir olmasıdır. Ancak, şifreleme ve şifre çözme için aynı anahtarın kullanılıyor olması dezavantaj doğurur. Tek bir anahtarın güvenliğinin sağlanması zordur. Diğer şahıslara bu anahtarın güvenli olarak gönderilmesi sorununun yanı sıra, bu şahısların anahtarı ne kadar gizli tutacağı sorun teşkil etmektedir. O nedenle, bu tür algoritmalar, daha çok paylaşımın olmadığı durumlar için uygundur. Bilgisayardaki dosyaların veya sabit diskin şifrelenmesi gerektiğinde kullanılabilirler.



Şekil 1.5. Simetrik Şifreleme

Simetrik sistemlerde anahtar dağıtma problemi:

Simetrik anahtarlı kriptografinin pratik kullanımındaki esas problem anahtar dağıtma problemidir. Bu problem temel olarak gönderici ve alıcının her ikisinin de anahtarın bir kopyasına sahip olmalarından kaynaklanır, bu ikili bir başkasının anahtarın bir kopyasını elde etmesini önlemelidir.

Varsayalım ki Mecnun ve Leyla, bilgiyi güvenlice değişmek, bilginin gizliliğini sağlamak için simetrik şifreleme sistemi kullanmak istesinler. Bu işlem için her ikisi de verinin şifrelenmesinde kullanılacak bir gizli anahtarı bilmelidir. Bununla birlikte, iletişim ortamı güvenilir olmadığından birbirleri ile görüşmelidirler. Bu, bir kez yapılabilirse; kullanıcılar gizli anahtarla şifrelenmiş, üçüncü kişilere anlamsız görünen bilgiyi mutlulukla değişebilirler. Fakat Mecnun ve Leyla kendilerine ait olan bu anahtarın güvenliğini bir kripto analizcisinin ele geçirip mesajları okuyabilme tehlikesine karşı korumalıdır.

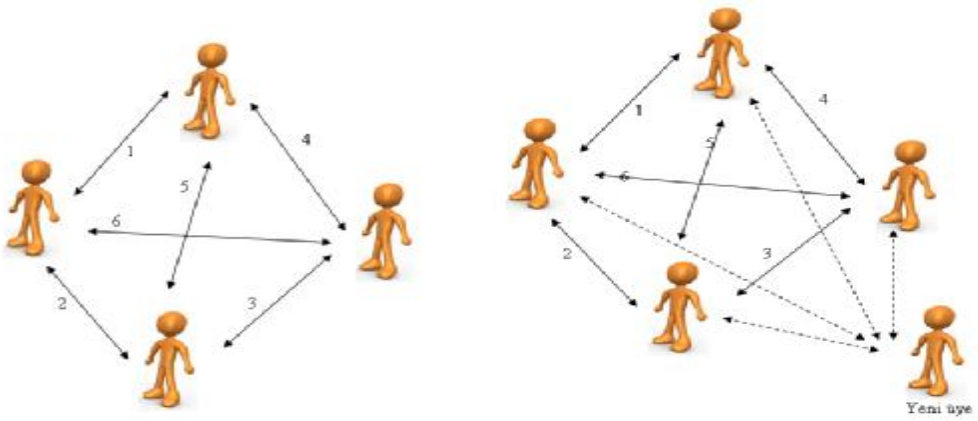
Eğer Mecnun ve Leyla başka biriyle, örneğin Ferhat ile yazışmak istesin, Ferhat'a anahtarı verirlerse onunla tam bir uzlaşma içinde olmalıdırlar çünkü kripto analizci anahtarı ele geçirebileceği yeni bir kaynağa daha sahip olmuştur. Tarafların birbirlerine gönderdiği her bir mesajın çözümlenmediğinden emin olmak için Mecnun ve Leyla, Ferhat ile iletişimde farklı anahtarlara sahip olmalıdırlar. Böylece her birinde iki anahtar bulunur. Mecnun, Ferhat'ın mesajını başka şekilde alır, Leyla'ya başka türlü iletir ve iletişim böyle sürer.

Şimdi 1000 üyeli bir sistem düşünelim, bunların tümü bir diğeri ile gizli bir iletişim kurmak istesin. Bu halde her bireyin iletişim kurduğu herkes için bir anahtara ihtiyacı olacaktır. Diğer bir deyişle diğer herkes 999 anahtara sahip olacaktır. Her birey de bu 999 anahtarı korumak zorundadır.

Bu şartlar dâhilinde *n* kullanıcılı sisteme yeni üye olanlara *n-1* tane anahtar verilir (Şekil 1.6). Buradan hareketle sistemde saklı tutulması gereken anahtar sayısı şöyle hesaplanır:

$$\text{Anahtar sayısı} = [n * (n - 1)] / 2 \qquad (1.1)$$

Saklanması gereken çok sayıda anahtar olacağından çoklu kullanıcılı ortamlar da asimetrik kripto sistemleri uygun çözümdür.



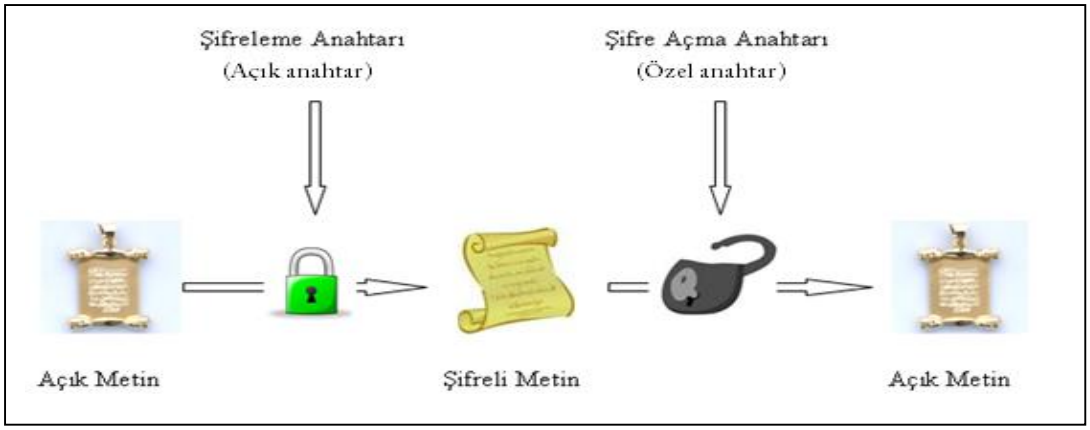
Şekil 1.6. Simetrik sistemde anahtar problemi (a) mevcut sistem (b) yeni üye katılımı

1.3.2.2. Asimetrik şifreleme

Asimetrik şifreleme algoritmaları simetrik şifreleme algoritmalarından radikal bir farklılık göstermektedir. (Çizelge 1.1) Bu tip şifreleme algoritmalarında açık (public) ve özel (private) anahtar olmak üzere iki ayrı anahtar kullanılmaktadır.

Açık anahtarlı algoritmalar da denilen asimetrik algoritmalar da şifreleme için kullanılan anahtar ile şifre çözme için kullanılan anahtar birbirinden farklıdır. Bu algoritmalara açık anahtarlı algoritmalar denmesinin sebebi şifre anahtarının halka (kamuya/genel kullanıma) açık olmasıdır. Anahtar çiftlerini üreten algoritmaların matematiksel özelliklerinden dolayı açık-gizli anahtar çiftleri her kişi için farklıdır, diğer bir deyişle her kullanıcının açık-gizli anahtar çifti yalnızca o kullanıcıya özeldir.

Bir kullanıcının gizli anahtarı, yalnızca kendi kullanımı içindir ve başkalarının eline geçmemesi gerekir. Bu kullanıcının açık anahtarı ise, bu şahsa mesaj göndermek isteyen herhangi biri tarafından kullanılabilir. Gönderici mesajı, alıcının açık anahtarı ile şifreler. Alıcı, gelen mesajı kendi özel anahtarı ile açar. (Şekil 1.7)



Şekil 1.7. Asimetrik Şifreleme

Asimetrik anahtarlı algoritmalarda önemli bir nokta da şifre çözüm anahtarının (en azından makul bir zaman dilimi içerisinde) şifre anahtarından hesaplanamaz olmasıdır.

Çizelge 1.1. Simetrik ve asimetrik şifreleme algoritmalarının genel özellikleri [8]

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*

Asimetrik algoritmalar, simetrik algoritmalara göre daha güvenli ve kırılması zor algoritmalardır. Bununla birlikte, performansları simetrik algoritmalara göre oldukça düşüktür. Özet olarak asimetrik şifreleme sisteminde gönderici ve alıcının gizli anahtarları paylaşmaları gereksinimi ortadan kalkmıştır. Tüm iletişimler sadece açık anahtar üzerinden gerçekleştirilir. Özel anahtarınız hiç bir şekilde paylaşılmaz ya da gönderilmez. Bu şifremle sistemi detaylı biçimde ikinci kısımda incelenecektir.

2. ASİMETRİK KRİPTOGRAFİ

Asimetrik (açık anahtarlı) kriptografi ile ilgili ilk düşüncelerin ortaya atılmasına kadar geçen süreçte, kullanılan simetrik kriptografi sistemler göz önünde bulundurulduğunda açık anahtarlı kriptografinin gelişmesi, bütün kriptografi tarihindeki en büyük devrimdir. İkinci dünya savaşında şifreleme/şifrele açma yapan rotor makinelerinin ortaya çıkması sonucunda, geleneksel kriptografide büyük bir gelişme kaydedildi ve bu gelişme, temelleri 1970'li yıllara dayanan asimetrik şifrelemenin önünü açmıştır.

Simetrik şifrelemedeki anahtar paylaşım sorununu çözmek için, 1976 yılında Whitfield Diffie ve Martin Hellman tarafından asimetrik şifreleme tekniği geliştirilmiştir. Bu yöntemde iki ayrı anahtar kullanılması ve herhangi bir anahtar transferinin gerekmemesi güvenliği artırmaktadır. Zaman içinde bu sisteme birçok algoritma önerilmiştir. Rivest, Shamir ve Adleman'ın meşhur RSA algoritması ve 80'li yıllarda parlamaya başlayan eliptik eğri tabanlı şifreleme (ECC) sistemleri en önemli olanlarıdır ve halen daha kullanılmaktadır.

Asimetrik kriptografi, gerçek anlamda daha önceki gelişmelerden radikal bir kopuştur. Asimetrik kripto sistemlerin en önemli noktaları matematiksel işlevler üzerine temellenmiş olmalarıdır. Aslında açık anahtarlı kriptografi matematiğin çözüm getiremediği bir takım durumları kullanarak güvenlik sağlar. Örneğin RSA siteminin güvenliği; matematikte çok büyük bir sayının, iki asal çarpanının bulunmasının herhangi bir doğrudan çözümü olmamasına dayanır.

Açık anahtarlı şifreleme sistemlerinin genel olarak iki ana kullanım alanı vardır: Şifreleme ve dijital imza.
Bu kullanımlar kabaca şu adımlarla gerçekleşir:

● Her ağdaki her son sistem, kullanıcı ya da benzeri, şifreleme ve şifre açma için kullanacak olduğu anahtar parçalarını yaratır.

● Her sistem, şifreleme anahtarını herkesçe erişilebilecek bir dosya ya da yazmaç içerisine kaydederek ya da duyurarak herkesçe erişilebilecek şekilde paylaşır. Bu anahtar, açık anahtardır. Özel anahtar saklı tutulur.

● Eğer, herhangi bir kullanıcı örneğin Mecnun, herhangi bir başka kullanıcıya örneğin Leyla, Leyla'nın bu mesajı kendisinden başka kimsenin

görüntüleyemediğine emin olabileceği bir mesaj yollamak isterse, mesajı Leyla'nın genel anahtarını kullanarak şifreler.

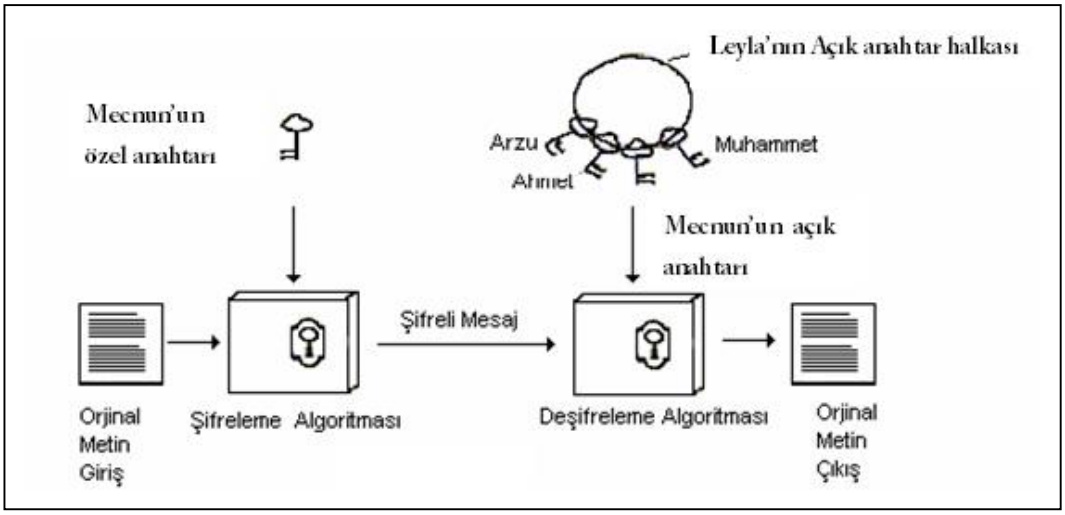
● Leyla, mesajı aldığında, bu mesajı kendi özel anahtarını kullanarak açar ve açık meten ulaşır. Diğer hiçbir alıcı mesajın şifresini açamaz, çünkü şifreyi çözecek olan özel anahtarı sadece Leyla bilir.

Yukarıdaki senaryo ile Leyla sadece kendisinin okuduğundan ve başka herhangi bir kimsenin görüntüleyemediğinden emin olduğu bir mesaj alır. Fakat bunun kimden geldiğinden emin olamaz. Gizlilik yani şifreleme sağlanmış olur.

Şekil 2.1' de gösterilen başka bir senaryo ise:

● Eğer, Mecnun; Leyla'ya mesajın kendisinden geldiğine emin olarak okuyabileceği bir mesaj yollamak isterse, mesajı kendisinin gizli anahtarını kullanarak şifreler.

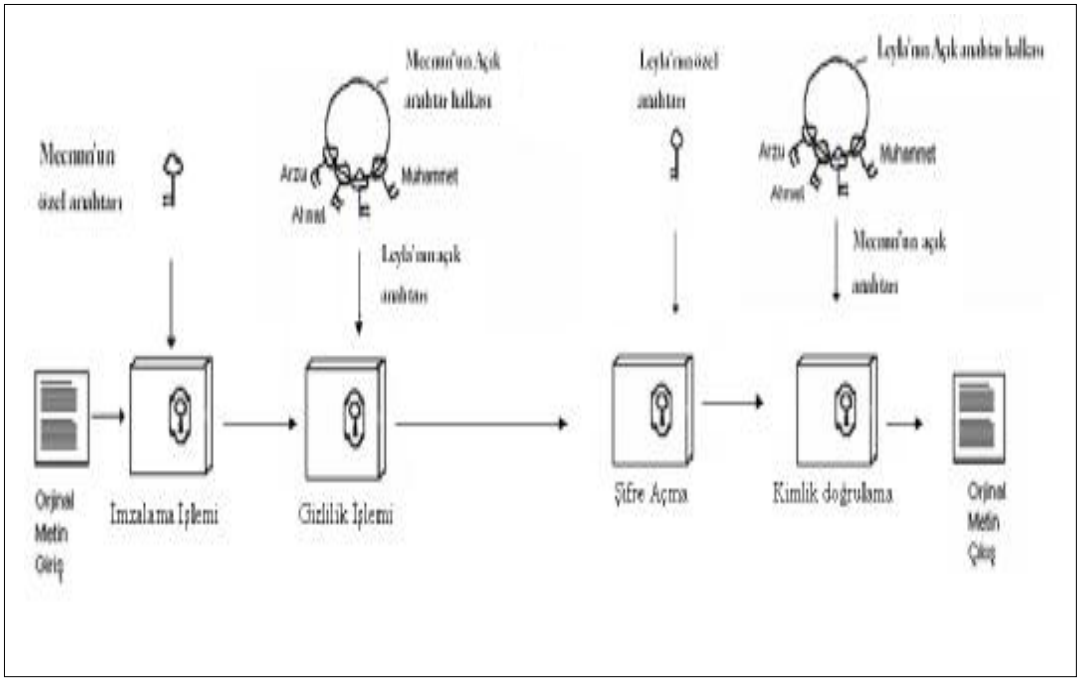
● Leyla, mesajı aldığında, bu mesajı Mecnun'un genel anahtarı ile deşifreler. Bu durumda Leyla, bu mesajın Mecnun'dan kendisine geldiğine ve kendisine ulaşana kadar yolda herhangi bir yerinin değiştirilmediğinden emin olur. Çünkü Mecnun'un genel anahtarı ile şifresini açtığı mesajın sadece Mecnun'un bilebileceği özel anahtar ile şifrelenmiş olabileceğini bilir. Bu senaryo ile de gizlilik yerine kimlik denetimi yani dijital imza sağlanmış olur. Fakat bu senaryoda şifre açma işlemini diğer alıcıların her biri de yapabilir, çünkü Mecnun'un genel anahtarı herkesçe bilinmektedir



Şekil 2.1. Asimetrik şifreleme dijital imza

Hem gizliliğin hem de kimlik denetiminin sağlanabileceği senaryo ise: (Şekil 2.2)

● Eğer, Mecnun; Leyla'ya, mesajın kendisinden geldiğine ve yolda başka kimsenin içeriğini görüntüleyemediğine emin olarak okuyabileceği bir mesaj yollamak isterse, mesajı kendisinin gizli anahtarını kullanarak şifreler, daha sonra ortaya çıkan mesajı da Leyla'nın genel anahtarını kullanarak şifreler. Bu sayede de hem gizlilik hem de iki taraflı kimlik denetimi sağlanmış olur.



Şekil 2.2. Asimetrik kriptografide imzalama ve gizlilik

2.1. Asimetrik Kriptografinin Avantajları

Bilgisayar bilim ve teknolojisinin eriştiği yüksek düzey göz önüne alındığında, simetrik kripto sistemlerin mutlak biçimde korumak zorunda oldukları anahtarların koruma ve dağıtım maliyetinin ne kadar yüksek ve koruma işleminin ne kadar zor olduğu kolayca görülebilir. Sırf bu nedenden ötürü, karşılıklı haberleşme içinde olan iki tarafın güvenli dağıtım kanalları oluşturması özellikle güncel bankacılık sisteminde, yaygın görülen bir örnektir [9]. Öte yandan, şifreleme ve şifrele açma dönüşüm fonksiyonlarının kullandıkları anahtarlar birbirinden ayrılarak anahtar güvenliği sorunu kesin biçimde çözülebilir. Asimetrik şifreleme sistemlerin tüm güvenliği şifre açma anahtarının yalnız ve yalnız yetkili alıcı tarafından bilinmesinde yatar. Öte yandan, her ne kadar her iki

anahtar birbirinden farklıysa da şifreleme anahtarından gidilerek şifre açma anahtarını oluşturmak, teorik olarak olası, ancak pratikte çözümsüz bir problemdir.

Asimetrik şifreleme sistemlerin, simetrik sistemlere göre önemli bir yararı da anahtar yönetimidir. Simetrik sistemlerde; *n* kişinin özel anahtar şifrelemesini kullanması durumunda grup içerisinde her kişi için bir farklı özel anahtar ihtiyacı olmaktadır. Böylece *n(n-1)* adet anahtar yönetimi olacaktır. Eğer *n* binlerce kullanıcı olursa o zaman milyonlarca anahtar yönetimi söz konusudur. Ayrıca gruba yeni bir kullanıcı eklemesi de kolay bir iş olmamaktadır, yeni kullanıcının gruptaki herkesle iletişim kurabilmesi için *n* adet yeni anahtar yönetimi söz konusu olacaktır. Daha sonra, yeni anahtarların gruba yollanması gerekmektedir. Aksine, asimetrik sistemlerde, kullanıcıların *n* adet genel anahtarları genel bir dizinde tutulur, yeni bir kullanıcı eklemesinde kullanıcının yeni genel anahtarı dizine eklenmesi yeterli olacaktır.

Asimetrik kripto sistemlerin en büyük avantajı, veri şifreleme ve şifre açma işlemlerini yapmasının yanında kimlik doğrulama, bütünlük, inkar edememezlik gibi prensipleri sağlayabilmesidir. (Çizelge 2.1)

Dijital imza gerçekleştirmede kullanılan asimetrik sistemler büyük bir ihtiyacı karşılamıştır. Aldığımız bir şifreli metnin şifre açma işlemini gerçekleştirdikten sonra karşılaşacağımız en büyük problem bu şifreli metnin bize doğru kişiden gelip gelmediğidir. Bu sorun da dijital imzalar sayesinde aşılabilmektedir.

Çizelge 2.1. Asimetrik ve simetrik kriptografi sistemlerinin özelliklerini karşılaştırma [10]

Konu	Simetrik Kriptografi	Asimetrik Kriptografi
Gizlilik	Sağlar	Sağlar
Bütünlük	--	Sağlar
Kimlik doğrulama	--	Sağlar
İnkâr Edememezlik	--	Sağlar
Performans	Hızlı	Yavaş
Güvenlik	Anahtar uzunluğuna bağlı	Anahtar uzunluğuna bağlı

Bireyleri tanımlama günümüz iletişiminde çok önemlidir. Konuştuğumuz kişinin bizi aldatmadığından, gerçek kişiyi taklit etmediğinden emin olmamız gerekir. Bu nedenle kişiler arasında bu işi yapabilecek tanımlayıcı bir protokol kullanır. Birçok sayıda tanımlayıcı protokol vardır ve bunlar genellikle RSA prensiplerine bağlıdırlar.

2.2. Asimetrik Kriptografinin Dezavantajları

Asimetrik şifreleme algoritmalarının da dezavantajları bulunmaktadır. Bu algoritmaların güvenliğini sağlayabilmek için çok büyük asal sayılar kullanılmaktadır. Bu da zaman açısından çok büyük problemler getirmektedir. Asimetrik bir algoritmayı kullanan sistemler, simetrik algoritmaları kullanan sistemlere göre çok daha yavaştır. Ayrıca asimetrik şifreleme algoritmalarının çok büyük sayılar kullanmasından dolayı donanımsal yapılara uyum sağlaması çok zor olmaktadır. Bundan dolayı bazı sistemler hem simetrik hem de asimetrik şifreleme algoritmalarını birlikte kullanarak, simetrik şifreleme algoritmalarının dezavantajı gizli anahtar güvenliği problemini ve asimetrik şifreleme algoritmalarının hız problemini ortadan kaldıramaya çalışmaktadır.

Kriptografi bilimi hızla gelişen bir bilim dalıdır. Eski algoritmaların dezavantajlarını ortadan kaldıracak yeni şifreleme algoritmaları sürekli gelişmektedir. Mesela asimetrik şifreleme algoritmalarının güvenliğinin temel prensibi olan çok büyük asal sayıları kullanımının yerine aynı güvenlik seviyesini daha düşük asal sayı değerleriyle gerçekleştirmeye çalışmaktadır.

Sonuç olarak, asimetrik şifreleme algoritmalarında ki hızlı gelişim, istenilen dezavantajları ortadan kaldırabilirse günümüz teknolojisinde simetrik şifreleme algoritmalarının yerini tamamen alabileceğini göstermektedir. Her iki şifreleme sistemi de günümüzde kullanılmaya devam etmektedir.

Bu sistemlerden birinin diğerine göre üstün olduğunu söylemek zordur. Asimetrik şifreleme algoritmaları ile yapılan işlemler (şifreleme, şifre açma, sayısal imzalama ve imza doğrulama işlemleri) çok değerli fakat yavaş işlemlerdir. Uygulamanın çalıştırıldığı platform, kullanılan algoritma ve anahtar uzunluğu işlemlerin hızını belirleyen önemli etkenlerdendir.

Uygulamaya göre sistem gereksinimleri göz önüne alınarak kullanılacak şifreleme yöntemi seçilmelidir. Asimetrik şifreleme algoritmaların tercih edilmesinin nedeni, sunduğu kripto analiz direnci, anahtar dağıtımdaki kolaylık, kimlik doğrulamaya imkan tanıması gibi nedenlerdir.

2.3. Asimetrik Kriptografi için Gereklilikler

Asimetrik sistemlerin temellerini atan Diffie ve Hellman, ilerde yazılması muhtemel asimetrik algoritmaların yerine getirmeleri gereken durumları şöyle sıralamışlardır:

1. Bir kullanıcı (B olsun) için, anahtar parçalarını (genel anahtar ve özel anahtar) yaratmak, hesapsal olarak kolay olmalıdır.
2. Gönderenin (A olsun), mesajı göndereceği kişinin (B olsun) genel anahtarını bildiği ve şifrelenecek olan mesajı (M) bildiği durumda, uygun şifreli metni (C) yaratmak hesapsal olarak kolay olmalıdır.

$$C = E_{KU_B}(M) \tag{2.1}$$

3. Alıcı B`nin, özel anahtarını kullanarak, şifrelenmiş mesajı orijinal haline getirmesi hesapsal olarak kolay olmalıdır.

$$M=D_{KR_B}(C)=D_{KR_B}(E_{KU_B}(M)) \tag{2.2}$$

4. Herhangi bir rakip için, genel anahtarı bilerek, özel anahtarı bulması hesapsal olarak imkânsız olmalıdır.
5. Herhangi bir rakip için, genel anahtarı, şifreli metini (C) bilerek orijinal mesajı (M) elde etmesi hesapsal olarak imkânsız olmalıdır.
6. Şifreleme ve şifre açma fonksiyonları her iki sıra ile de uygulanabilir olmalıdır.

$$M=E_{KU_B}(D_{KR_B}(M)) \tag{2.3}$$

Asimetrik kriptografinin en önemli dayanak noktası tersine ise çevrilebilir fonksiyonun (f_k) bire-bir olduğu bir aralıkta, fonksiyonun kendisinin hesaplanması kolay iken fonksiyonun tersini hesaplamak imkânsızdır.

$Y = f_k(X)$ k ve X biliniyorsa, kolay...

$X = f^{-1}{}_k(Y)$ k ve Y biliniyorsa kolay...

$X = f^{-1}{}_k(Y)$ Y biliniyor fakat k bilinmiyorsa çözülemez [11].

2.4. Algoritmaların Karşılaştırılması

Kriptografi biliminde birçok farklı algoritma ileri sürülmüştür. Bu algoritmaların birçoğu geliştirilerek kullanıma sunulmuştur. Bu algoritmaları yaptıkları iş açısından karsılaştırmak için her algoritmaya uygulanabilecek somut ölçüler tanımlanmalıdır. Aynı işi yapan algoritmalardan daha az işlemde sonuca ulaşanın (hızlı olanın) belirlenmesi yani daha genel olarak algoritma analizi teorik bilgisayar bilimlerinin önemli bir alanıdır.

Bir programın performansı genel olarak programın işletimi için gerekli olan bilgisayar zamanı ve belleğidir. Bir programın zaman karmaşıklığı (time complexity) programın işletim süresidir. Bir programın yer karmaşıklığı (space complexity) programın işletildiği sürece gerekli olan yer miktarıdır. Bir problemin çözümünde, kullanılabilecek olan algoritmalardan en etkin olanı seçilmelidir. En kısa sürede çözüme ulaşan veya en az işlem yapan algoritma tercih edilmelidir. Burada bilgisayarın yaptığı iş önemlidir. Bazı durumlarda da en az bellek harcayan algoritmanın tercih edilmesi gerekebilir. Biz bu bölümde algoritmaları zaman bakımından inceleyeceğiz.

2.4.1. Zaman karmaşıklığı

Algoritmaların karşılaştırılmasında genellikle asimptotik etkinlikleri dikkate alınır. Girdi boyutu sonsuza yaklaşırken işletim süresinin artışı, Asimptotik gösterimin elemanı olan 4 önemli gösterim ile hesaplanır: O-notation, o-notation, Ω-notation, θ-notation. Burada sadece Büyük-O gösterimi üzerinde durulacaktır. Büyük-O gösterimi, fonksiyonların artış oranının üst sınırını belirler. Büyük-O gösterimi bilgisayar bilimcileri tarafından algoritmaları davranışlarını tanımlamak için kullanılır.

Büyük-O Gösterimi (notasyonu):

n elemanlı bir listedeki elemanların toplamını bulmak için *n-1* toplama işlemi yapmak gerekir. Bu şekilde yapılan iş, girdi boyutunun bir fonksiyonu olarak ele alınmış olur. Bu fonksiyon yaklaşımını matematiksel gösterim kullanarak ifade edebiliriz: Büyük-O gösterimi veya büyüklük derecesi (order of

magnitude). Büyüklük derecesini problemin boyutuna bağlı olarak fonksiyonda en hızlı artış gösteren terim belirler.

Örnek olarak:

Eğer bir *f*(*n*) işlevi diğer işlevlerin sonlu toplamı olarak yazılabiliyorsa o zaman bunların içinden en hızlı büyüyeni *f*(*n*) işlevinin derecesini belirler

$$f(n) = 10 \log n + 5 (\log n)^3 + 7n + 3n^2 + 6n^3 \in O(n^3)$$

$$f(n) = n^4 + 100n^2 + 10n + 50 \in O(n^4)$$

Özel olarak eğer bir işlev *n* terimine bağlı birçok terimli tarafından üstten sınırlandırılabiliyorsa o zaman *n* değeri sonsuza gittikçe çok terimlinin *düşük dereceli* terimleri ihmal edilebilir.

Örneğin ikinci fonksiyonda *n*'in derecesi n^4'tür yani *n*'in büyük değerleri için fonksiyonu en fazla n^4 etkiler. *n*'in çok büyük değerleri için n^4; $100n^2$'den, *10n*'den ve *50*'den çok büyük olacağından daha düşük dereceli terimler dikkate alınmayabilir. Bu diğer terimlerin, işlem süresini etkilemedikleri anlamına gelmez; bu yaklaşım yapıldığında *n*'in çok büyük değerlerinde diğer terimlerin önem taşımadıkları anlamına gelir.

Fonksiyonlardaki Büyük-O değerleri şu şekilde özetleyebiliriz.

Toplama

$$o(f(n)) + o(g(n)) = o(\max\{f(n), g(n)\}) \qquad (2.4)$$

Çarpma

$$o(f(n)).o(g(n)) = o(\{f(n), g(n)\}) \qquad (2.5)$$

RSA'nın analizini yaparken kullanacağımız değerleri bulalım. İkili sistemde toplama ve çarpmadaki işlem sayılarını bulalım.

Toplama

Bir toplama örneği ele alalım.

```
1010 +
 101
-------
1111
```

k bit iki sayının toplanması *k bit* işlem gerektirir.

$m > k$ iken *m bit* bir sayı ile *k bit* bir sayının toplanması *k bit* işlem gerektirir.

Çarpma

Bir çarpma örneğini ele alalım.

M = 11101 ve N=1101

```
        11101 * (k   bit)
         1101   (l   bit)
-----------------
        11101    (sıra 1)
       00000     (sıra 2)
      11101      (sıra 3)
     11101       (sıra 4)
---------------
  101111001
```

Bu çarpma işleminde önce l *bit* kadar sıramız oluşuyor. Oluşan her sıranın uzunluğu ise k *bit* kadardır. Her bir sıra toplanırken k *bit* kadar işlem gerekir. Bu nedenle çarpma işleminde $k * l$ kadar işlem gerekir [12].

Artış Oranı Fonksiyonları

O(1) : Sabit zaman

Örnek: n elemanlı bir dizinin i. elemanına bir değer atanması O(1)'dir. Çünkü bir elemana indisinden doğrudan erişilmektedir.

O(n) : Doğrusal zaman

Örnek: n elemanlı bir dizinin tüm elemanlarının ekrana yazdırılması O(n)'dir.

Örnek: Sıralı olmayan bir dizideki (listedeki) elemanlardan birinin aranması O(n)'dir

$O(\log_2 n)$: O(1)'den fazla O(n)'den azdır.

Örnek: Sıralı bir listenin elemanları içinde ikili arama (binary search) uygulanarak belirli bir değerin aranması $O(\log_2 n)$'dir.

$O(n^2)$: İkinci dereceli zaman

Örnek: Basit sıralama algoritmalarının birçoğu (selection sort gibi) $O(n^2)$'dir.

$O(n \log_2 n)$: Bazı hızlı sıralama algoritmaları $O(n \log_2 n)$'dir.

$O(n^3)$: Kübik zaman

O(2^n) : Üstel zaman, çok büyük değerlere ulaşır [13].

Örnek süre hesaplaması:

• Bir programın işletimi n^3 adım sürüyorsa ve n=1000 ise, program 1000^3 adım sürecek demektir. Yani 1 000 000 000 (bir milyar) adım.

• Kullanılan bilgisayar saniyede 1 000 000 000 adımı gerçekleştirebilecek kadar hızlı ise bu program tam 1 saniye sürecektir.

2.5. Sayı Teoremi

Asimetrik kripto sistemlerinin çoğunluğu, sayılar teorisini temel almıştır. Bu tezdeki sonuçları algılamak için, sayılar teorisini biliyor olmaya çok da gerek yoktur. Bununla birlikte, açık anahtarlı şifreleme algoritmaları hakkında kesin bir yargıya varmak için, sayılar teorisinin bazı kısımları hakkında bilgi sahibi olmak faydalı olabilir.

2.5.1. Çinli kalan teoremi (Chinese Remainder Theorem (CRT))

Bu teorem Çinliler tarafından ortaya çıkarılmıştır. Farklı "mod" değerinden kalanı bilenen sayının kolayca hesaplanmasını sağlar.

Örnek ile açıklarsak:

x sayısının 3,5,11,16 sayılarından kalanı aşağıdaki gibi olsun.

x== 2 mod 3,
x== 3 mod 5,
x== 4 mod 11,
x==5 mod 16.

Bu verilerden Çin kalan teoremini kullanarak x değerini hesaplayacağız.

Öncelikle her bir "mod" değeri için diğer mod değerlerinin çarpımı hesaplanır.

M= 3*5*11*16 = 2640.
M_1 = 2640/3 = 880,
M_2 = 2640/5 = 528,
M_3 = 2640/11 = 240,
M_4 = 2640/16 = 165.

Sonra bulunan bu değerlerin mod üzerinden tersi hesaplanır. Bu işlemler için Euclidean algoritması (Kısım 2.5.2) kullanılabilir.

$880^{-1} \bmod 3$ è $880*N_1 == 1 \bmod 3$ è $N_1 = 1$,
$520^{-1} \bmod 5$ è $520*N_2 == 1 \bmod 5$ è $N_2 = 2$,
$240^{-1} \bmod 11$ è $240*N_3 == 1 \bmod 11$ è $N_3 = 5$,
$165^{-1} \bmod 16$ è $165*N_4 == 1 \bmod 16$ è $N_4 = -3$.

Sonuç olarak $x = 2*880*N_1 + 3*528*N_2 + 4*240*N_3 + 5*165*N_4$

$= 2*880*1 + 3*528*2 + 4*240*5 + 5*165*(-3)$

$= 7253 \bmod 2640$

$= 1973$

Çin Kalan Teoremini şu şekilde de yorumlayabiliriz. Bir sayının 2640 mod değerinden kalanı bulabilmek için 2640 sayısını oluşturan çarpanların (3,5,11,16) mod'undan kalanı bilmemiz yeterlidir.

Buradan hareketle önce Gauss, RSA sistemi için bir düzenleme yapmış:

n=p.q; d= özel anahtar ve m= mesaj iken C= $m^d \bmod n$ işlemi şu şekilde ikiye ayrılabilir:

$$C_p = (m \bmod p)^{d \bmod (p-1)} \bmod p \quad (2.6)$$

$$C_q = (m \bmod q)^{d \bmod (q-1)} \bmod q \quad (2.7)$$

C=CRT(C_p, C_q)

C = $[(C_p \times q \times \underbrace{(q^{-1} \bmod p)}) + C_q \times p \times \underbrace{(p^{-1} \bmod q)}] \bmod n$

= $[C_p \times \quad Xp \quad + C_q \times \quad Xq \quad] \bmod n$

Daha sonra Garner düzenleme yaparak:

C= CRT(C_p, C_q)

= $C_p + p.\underbrace{[(C_p - C_q) \times (p^{-1} \bmod q) \bmod q]}$

= $C_p + p. \quad v$ (2.8)

sonucuna ulaşmıştır. Bizde RSA'da bu düzenlemeyi kullanacağız [14-20].

2.5.2. Euclidean algoritması

Bu algoritmada sayıların ortak bölenleri kolayca bulunur. RSA sisteminde daha çok sayıların, ortak böleni bulmak için değilde *mod n'* e göre tersini bulmak için Euclidean Algoritmasını kullanacağız.

Örnek 1:
81 ile 57'nin ortak bölenini hesaplayalım. Öncelikle iki sayı a=b.x + y şeklinde dönüştürülür.

81 = 1(57) + 24

Sonra ise bir önceki bölen ile kalan aynı sekle dönüştürülür bu durum bulana kadar devam eder. Sıfır kalanını veren sayı en büyük ortak bölendir.

57 = 2(24) + 9
24 = 2(9) + 6
9 = 1(6) + 3
6 = 2(3) + 0 **è** OBEB (87,51)=3

Örnek 2:
1239 ile 735'in ortak bölenini bulalım.

1239 = 1 (735) + 504
735 = 1 (504) + 231
504 = 2 (231) + 42
231 = 5 (42) + 21
42 = 2 (21) +0 **è** OBEB(1239; 735) = 21

Bu algoritma üzerinde bazı yapılan çalışmalar yapıldı ve bu sayede bir sayının belirli bir mod değerine göre tersi hesaplanmaya başlandı. Bize bu RSA çalışmada lazım olacak bu algoritmayı bir örnek ile inceleyelim:

Örnek 3: $15^{-1} \bmod 26$ değerini hesaplayalım.

Step 0: 26 = 1(15) + 11 $p_0 = 0$
Step 1: 15 = 1(11) + 4 $p_1 = 1$
Step 2: 11 = 2(4) + 3 $p_2 = 0 - 1(1) \bmod 26 = 25$
Step 3: 4 = 1(3) + 1 $p_3 = 1 - 25(1) \bmod 26 = -24 \bmod 26 = 2$
Step 4: 3 = 3(1) + 0 $p_4 = 25 - 2(2) \bmod 26 = 21$
$p_5 = 2 - 21(1) \bmod 26 = -19 \bmod 26 = 7$

$15^{-1} \bmod 26 = 7$ [14-20]

2.5.3. Hensel lifting teoremi

Kurt Hensel'in teoremi *M mod p* değerini biliyorken *M mod* p^2 değerine kolayca ulaşabilmeyi sağlayan bir sayı teoremidir.

p-adik sayı yaklaşımına göre $M \bmod p^2 = M \bmod p + pM_1$

$M_0 = M \bmod p$ ve $X_p = pM_1$ olsun.

$$C = M^e \bmod p^2$$
$$= (M_0 + X_p)^e \bmod p^2$$
$$= M_0^e + eM_0^{e-1}X_p \bmod p^2$$

$$X_p = (C - M_0^e)\,((eM_0^{e-1})^{-1} \bmod p) \bmod p^2$$
$$= (C - M_0^e)\,(e^{-1} \bmod p)\,(M_0^{1-e} \bmod p) \bmod p^2$$
$$= (C - M_0^e)\,(e^{-1} \bmod p)\,(C^{dp-1} \bmod p) \bmod p^2$$

$$M \bmod p^2 = M_0 + (C - M_0^{\,e})(e^1 \bmod p)(C^{dp-1} \bmod p) \bmod p^2 \qquad (2.9)$$

Bu denklem Kuvvet RSA'nın şifre açma kısmının çok daha hızlanmasına önayak olmuştur [20].

3. RSA (RON RIVEST, ADI SHAMIR, LEN ADLEMAN) ŞİFRELEME

RSA şifreleme sistemi açık anahtarlı şifreleme sistemlerinin en bilinenlerindendir. Ron Rivest, Adi Shamir ve Len Adleman tarafından 1977 yılında geliştirilmiştir ve geliştiricilerinin soyadlarının baş harfleri olan RSA olarak anılmaktadır.

RSA şifreleme sisteminin oluşturulmasıyla birlikte asimetrik şifreleme algoritmalarının günümüzde daha yaygın olarak kullanılması sağlanmıştır. RSA kripto sistemi, hem gizlilik hem de dijital imza sağlamak amacıyla kullanılabilir. Bu iki işlemi birden yapabilen ilk sistemdir. Günümüzde de RSA şifreleme algoritması halen daha güvenilirliğini korumaktadır. Bunun nedeni modüler matematik üstüne kurulmuş, kripto analizi asal sayılara, çarpanlara ayırmaya dayalı anlaşılması kolay ama çözülmesi zor bir algoritma olmasıdır.

Tipik bir asimetrik şifreleme algoritması olan RSA kripto sisteminde kişilere şifreli mesaj gönderilebilmesi için o kişilerin açık anahtarlarına ihtiyacı vardır. Mesajı alan kişinin de mesajı okuyabilmesi için gizli bir anahtarın olması gerekir.

3.1. RSA Sisteminin Güvenliği

RSA sistemine en çok zarar verecek saldırı bir kripto analistin belli bir açık anahtara karşı gelen gizli anahtarı bulmasıdır. Bunu başarabilen bir hasım hem şifrelenen bütün masajları okuyabilir, hem de imzaları taklit edebilir. Bunu yapmanın en akla gelen yolu N'nin asal çarpanlara ayrılması yani p *ve* q'nun hesaplanmasıdır. p, q ve açık üs e kullanılarak özel anahtar d kolaylıkla hesaplanabilir. Ancak buradaki zorluk N modülünün çarpanlarına ayrılmasıdır. RSA sisteminin güvenliği çok büyük sayıların asal çarpanlarına ayrılmasının zorluğu varsayımına dayanır.

Büyük sayıların çarpanlarına ayrılmasının zorluğu ispatlanmış değildir. Son üç yüzyıl içerisinde birçok ünlü matematikçiler bu konuda çalışmalar yapmışlardır. Fakat bu konuda belirlenebilmiş bir algoritma yoktur.

Şifreleme için seçilen p ve q asal sayıların çarpımlarından oluşan N sayısının boyutu, RSA algoritmasında anahtar boyu (key size) olarak anılır. Anahtarın boyutu büyüdükçe ilgili anahtarla şifrelenmiş metnin şifre çözme anahtarına sahip olmayan kişiler tarafından çözülmesi de zorlaşır.

RSA Çarpanlara Ayırma Problemi ilk olarak RSA Security Şirketi tarafından Mart 1991'de başlatılmıştır. En etkileyici sonuç RSA-155 (155 haneli anahtar) ile alınmıştır. RSA-155 çağrısı duyurulduktan yedi ay sonra Ağustos 1999'da bir grup araştırmacı tarafından 300 iş istasyonu ve PC'ler kullanılarak bu görevi tamamlamıştır. 512-bit şifrenin 1995 yılında 1 milyon $ dan bir yatırımla sekiz ayda kırılabileceği ileri sürülmüştü. Fakat RSA-155 ancak 1999 yılında yedi ayda kırılabildi. 512-bit olan bu şifrenin çözülmesi önemliydi çünkü o yıllarda internet üzerinden yapılan e-ticaret uygulamalarında 512-bit'lik şifreleme kullanılıyordu.

RSA Security şirketi 2010 yılı içinde 768 bitlik şifrenin 6 ayda 100.000 iş istasyonu ile çözüldüğünü duyurdu. Verinin değerine ve değerini koruma süresine göre 768 bitlik şifrelerin halen daha kullanılabileceğini belirtti. Bununla birlikte, RSA Security şirketi 1024-bit RSA modülünün çözülmesinin 768-bit modülüne göre bin kat daha zor olduğunu açıkladı ve 1024 bitlik bir şifrenin 10 yıl sonra çözülebileceğini düşünmenin bile çok iyimser olacağını belirtti [21].

N sayısının çarpanlarına ayrılabilmesi tehdidinin yanında “seçilmiş metin saldırısı” da tüm asimetrik sistemlere olduğu gibi RSA’ya karşıda kullanılabilir. Asimetrik şifreleme sistemlerinde saldırgan herhangi bir kullanıcının açık anahtarını görebilir. Buradan yola çıkan saldırgan, açık anahtar ile kendi belirlediği bir mesajı şifreler ve daha sonra bu mesaj üzerinde anahtar uzayının olası bütün özel anahtarlarını deneyerek şifreyi açmaya, kendi belirlediği mesaja ulaşmaya çalışır. Bu durumda RSA’nın güvenliği yine N sayısı ile doğru orantılıdır. N sayısının büyüklüğü anahtar uzayının da büyüklüğünü belirlemektedir.

3.2. RSA'nın Matematiği

1) M<N olduğu koşulda, $M^{ed} = M \bmod N$ iken, *e*, *d*, *N* değerlerini bulmak mümkün olmalıdır.

2) M<N koşulunu sağlayan tüm *M* değerleri için

$$C = M^e \bmod N \tag{3.1}$$

iken *C* ve $C^d \bmod N$ değerlerinin hesaplanması nispeten kolay olmalıdır.

3) Yalnız *e ve N* verildiğinde, *d* değerinin hesaplanması imkânsız olmalıdır.

Yukarıdaki bilgiler doğrultusunda aşağıdaki form için bir ilişki bulmalıdır:

$$M^{ed} = M \bmod N \tag{3.2}$$

Euler'in teoremine göre, verilen iki asal sayı *p ve q*,

$$N=p.q \tag{3.3}$$

ve 0<M<N olduğu durumda keyfi seçilmiş bir *k* tamsayısı seçilmiş diğer sayılar ile şöyle bir ilişki oluşturur:

$$m^{k\Phi(n)+1} = m^{k(p-1)(q-1)+1} \equiv m \bmod N \tag{3.4}$$

Burada bahsi geçen $\Phi(n)$ fonksiyonunun döndürdüğü değer, *n* değerinden küçük olan ve *n* ile aralarında asal olan tam sayıların sayısıdır. *p ve q* asal sayılar olduğu durumda,

$$\Phi(n) = (p-1)(q-1) \text{ olur.} \tag{3.5}$$

Böylece aşağıdaki eşitlikte istenen ilişkiye ulaşılır:

$$ed = k\Phi(n)+1 \tag{3.6}$$

Bu durumda da aşağıdaki denkliklerden bahsetmek mümkün olur:

$$ed \equiv 1 \bmod \Phi(n) \tag{3.7}$$

$$d \equiv e^{-1} \bmod \Phi(n) \tag{3.8}$$

e ve d; *mod* Φ (n) fonksiyonunun çarpmaya göre tersidir. Modüler aritmetiğin kurallarına göre bu denkliğin doğru olması *d*'nin *ve e'nin;* $\Phi(n)$ ile aralarında asal olması durumunda mümkün olabilir.

$$OBEB(\Phi(n), d) = 1 \text{ ve } OBEB(\Phi(n), e) = 1. \tag{3.9}$$

3.2. RSA'nın Çalışma Sistemi

Aşağıda açıklanan RSA şifreleme algoritmasının çalışması Şekil 3.1.'de gösterilmiştir.

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - *n/2* bit büyüklüğünde iki asal sayı *p ve q* seçilir.
  - İki asal sayının *(p ve q)* çarpımından *n bit* büyüklüğünde *N* sayısı hesaplanır.
  - Denklem (3.5) ile hesaplanan Φ (n) ile ortak böleni bir olan ve 1< e < Φ (n) koşulunu sağlayan açık anahtar *e* sayısı seçilir.
  - 1< d < Φ (n) koşulunu sağlayan özel anahtar *d* (3.6) denkliğinden hesaplanır.
- Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.
- Bu değerlerden *(d,n)* özel anahtar olarak elde tutulur. *d* değeri kimse ile paylaşılmaz.

Şifreleme:

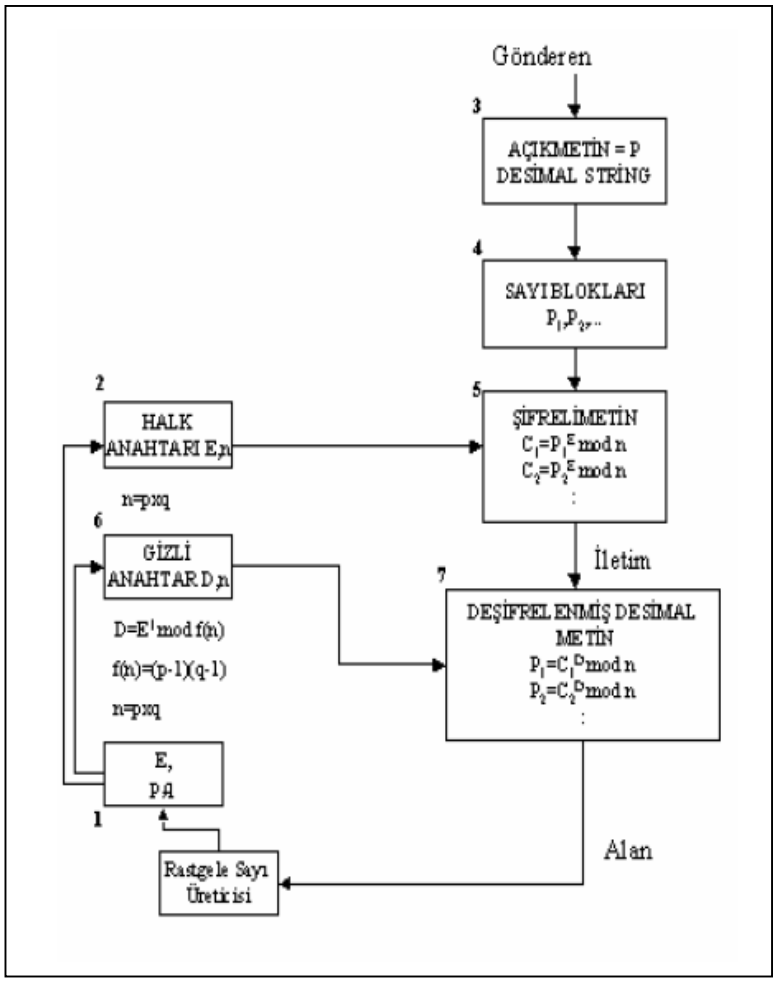
- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin (*M)* bloklarını şifreler ve elde etiği şifreli değerleri (*C)* karşıya gönderir.

$$C = M^e \bmod N$$

Şifre açma:

- Şifreli mesajı alan kullanıcı özel anahtarı ile şifreli değeri açar ve gerçek mesaja ulaşır [22].

$$M = C^d \bmod n$$



Şekil 3.1. RSA'nın çalışma mantığı [8]

Örnek:

"Kriptosistem" yazısının RSA ile şifrelenmesi Şekil 3.2 de gösterilmiştir. Bu işlemin basamaklrı şu şekildedir:

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - İki asal sayı *p=73 ve q=151* seçilmiş.
  - İki asal sayının *(p ve q)* çarpımından *N=7x151=11023* sayısı hesaplanır.
  - Denklem (3.5)'den Φ (n) =72x150=10800. Φ (n) ile ortak böleni bir olan açık anahtar *e=11* sayısı seçilmiş.
    1< e < Φ (n)

- $d \equiv e^{-1} \bmod \Phi(n)$ denkliğinden özel anahtar *d=5891* hesaplanmış. $1< d < \Phi(n)$

ü *(e=11,N=11023)* açık anahtar olarak yayınlanır.

ü *(d=5891,N=11203)* özel anahtar olarak elde tutulur. *d* değeri kimse ile paylaşılmaz.

Şifreleme:

ü Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürmüş.

KR	IP	TO	SI	ST	EM
1320	1119	2317	2111	2123	0513

ü Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin (*M)* bloklarını şifreler ve elde etiği şifreli değerleri (*C)* karşıya gönderir.

$C = M^e \bmod n$

$C_1 = 1320^{11} \bmod 11023$ **è** 10124 ; $C_2 = 1119^{11} \bmod 11023$ **è** 5618; …

Şifre açma:

ü Şifreli mesajı alan kullanıcı özel anahtarı ile şifreli değeri açar ve gerçek mesaja ulaşır.

$M = C^d \bmod n$

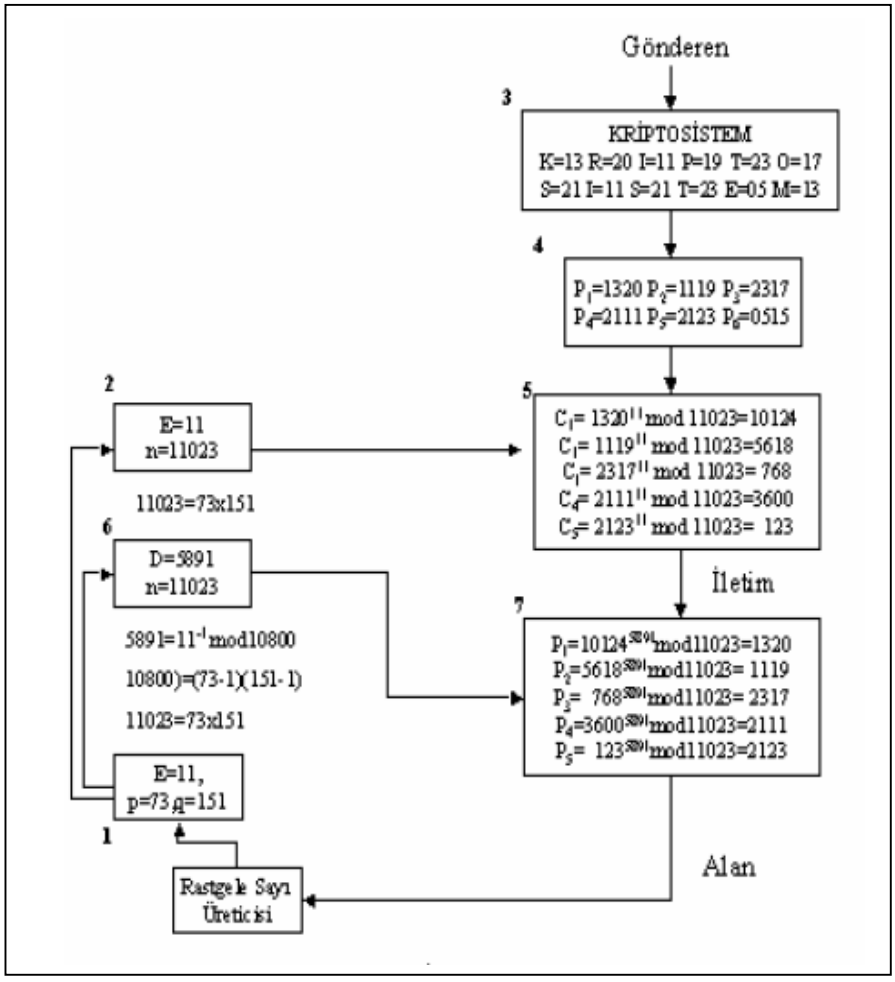
$M_1 = 10124^{11} \bmod 11023$ **è** 1320 **è** KR ; …..

3.3. RSA'nın Büyük-O Analizi

RSA da şifreleme ve şifre açmadaki ana işlem olan $C^d \ mod \ N$ işlemini irdeleyelim:

t **è** *d* sayısının bit uzunluğu ve *n* **è** mod sayısının (*N*) bit uzunluğu iken $C^d \ mod \ N$ değerinin hesaplanmasında *n bitlik* sayıların çarpma işlemi *t bit* kadar tekrarlanacaktır. Çarpma işleminin O(n*n) olduğu kısım 2.4.1. belirtilmişti. Bu işlem *t bit* kadar tekrarlanacak: $O(t.n^2)$

RSA algoritmasında $t \cong n$ bu yüzden RSA'nın şifre açma ve şifreleme zaman analizi için $O(n^3)$ olarak belirtebiliriz [23].



Şekil 3.2. “Kriptosistem” kelimesinin RSA ile şifrelenişi [8]

4. ÖNERİLEN RSA ALGORİTMASI

Teknolojinin ve bilimin sürekli geliştiği düşünüldüğünde; RSA'nın da gelişen kriptografi dünyasına uyum sağlayabilmesi, bu dünyada ayakta kalabilmesi için gelişmesi, yeniliklere uğraması kaçınılmazdır. Bu kısımda bu konuda yapılan çalışmalar irdelenerek, yeni bir RSA algoritması önerilmiştir. RSA ile ilgili yapılan geliştirilmeler daha çok şifre açma kısmının hızlanmasına yöneliktir. Şifreleme kısmı ile ilgili fazla bir çalışma bulunmamaktadır. Bu tezde RSA'nın şifreleme kısmını verimli hale getiren ama bunu yaparken şifre açma kısmını yavaşlatan Verimli RSA Şifreleme Şemasının hızlandırılması için bir algoritma önerilmiştir.

4.1. RSA Geliştirmeleri

RSA'nın geliştirilmesi için yapılmış olan çalışmaların önemlileri; şifreleme kısmını ilgilendiren ile şifre açma kısmını ilgilendirenler olmak üzere iki kısımda incelenmiştir.

4.1.1. Verimli RSA Şifreleme (encryption) Şeması

RSA'nın şifreleme kısmını ilgilendiren bu geliştirme ile 2008 yılında daha geniş anahtar havuzu yaratan bir sistem geliştirildi. Aynı zaman da bu geliştirdikleri sistem ile "Hill Cipher (matris ile şifreleme)" şifreleme sisteminde h^2 *(h:şifreleme sistemi içinde seçilen bir sayı)* tane bloğu matris ile gönderilebilecek hale getirdiler [24].

Verimli RSA Şifreleme Şemasının çalışma sistemi:

Bu sistem diğer sistemlerden farklı olarak; açık ve özel anahtarın (*e ve d*), Φ (n) değerine üzerinden değil de doğrusal cebir yaklaşımı ile oluşturulmuş olan *g* sayı üzerinden bulunmasını öngörür.

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
    - *n/2 bit* büyüklüğünde iki adet asal sayı *p ve q* belirlenir.

- Bu asal sayıların çarpımından *n bit* büyüklüğünde *N* sayısı hesaplanır. $N=p.q$
- Sonra bir *h* sayısı belirlenir. Doğrusal cebir yaklaşımı ile *g* sayısı oluşturulur.

  $g=(p^h-1)(p^h-p)(p^h-p^2)....(p^h-p^{h-1}).(q^h-1)(q^h-q)(q^h-q^2)....(q^h-q^{h-1})$ *(4.1)*
- OBEB (g, e) =1 koşullunu sağlayan açık anahtar *e* seçilir.
- $d \equiv e^{-1} \bmod g$ denkliğinden özel anahtar *d* belirlenir.

ü Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

ü Bu değerlerden *(d,n)* özel anahtar olarak elde tutulur.

Şifreleme:

ü Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür

ü Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin (*M)* bloklarını şifreler ve elde etiği şifreli değerleri (*C)* karşıya gönderir.

$C = M^e \bmod N$

Şifre açma:

ü Şifreli mesajı alan kullanıcı özel anahtarı ile şifreli değeri açar ve gerçek mesaja ulaşır.

$M = C^d \bmod N$

Verimli RSA Şifreleme (encryption) Şemasının Büyük-O analizi

Bu sistem zaman karmaşıklığına oldukça büyük bir yük bindirmektedir. Anahtar uzayını arttırmak için kullanılan *g* değeri e *ve d* değerlerini oldukça büyüttüğü için sistem oldukça yavaşlamaktadır.

p ve q; n/2 bit iken *N* sayısı *n bit,* g sayısı ise $n.h^2$ *bittir*. Açık anahtar ile özel anahtar da *g* sayısından belirlendiğinden *e* (açık anahtar kullanıcı tarafından seçilirse daha düşük olabilir) *ve d* de *n.h bittir*.

RSA'nın şifre açma kısmı O (n^3) iken bu şemada O ($n^3.h^2$) dır. Bu da şifre açma kısmının Standart RSA'ya göre çok yavaşlaması anlamı taşımamaktadır. *h* değeri yükseldikçe hız açısından verimlilik daha da düşecektir. Bu aşırı yavaşlama getirilerine rağmen bu şemanın tercih edilmeme yüzdesini yükseltmektedir.

Aynı şekilde bu sistemin şifreleme kısmında yavaş olabileceğini söyleyebiliriz. Eğer açık anahtar $n.h^2$ *bitlik* havuzdan yüksek seçilirse aynı oranın şifreleme kısmı içinde olacağını söyleyebiliriz. Fakat şifreleme kısmının yavaşlamaması adına açık anahtarın küçük seçilmesi güvenlik zaafı oluşturmadan sistemin hızını koruyacaktır.

RSA'nın şifreleme kısmını fazla yormayan fakat şifre açma kısmına yük bindiren bu sistemin şifre açma kısmını geliştirebilmek için bu yönde yapılan çalışmalar bir sonraki kısımda ele alınmıştır.

4.1.2. RSA şifre açma (decryption) geliştirmeleri

RSA algoritması, tüm asimetrik sistemler gibi simetrik sistemlere nazaran daha yavaş bir şifreleme sistemidir. Bu yüzden özellikle RSA'nın şifrele açma kısmını hızlandırmak yeni algoritmalar geliştirilmiştir. Bu kısımda bu algoritmalar ele alınmıştır.

4.1.2.1. Çinli kalan teoremi (Chinese Remainder Theorem) ile RSA (CRT RSA)

CRT RSA, şifre açma işlemini hızlandırmak için en çok kullanılan, en yaygın yöntemdir. İlk olarak Couvreur ve Quisquater tarafından 1982 de tanımlanmıştır. Bu sistem; şifre açma kısmını daha küçük mod işlemlerine dayandırarak çalışmayı prensip eder [25,26]. Bu metot tek ve büyük sayılarla yapılan mod işlemini; Çin Kalan Teoremi yardımıyla iki kısma ayırıp daha küçük sayılarla daha hızlı çalışmayı öngörür.

CRT RSA'nın çalışma sistemi:

Anahtar üretimi ve şifreleme kısmı standart RSA gibi çalışır. Şifre açma kısmında ise Çin Kalan Teoremi devreye girer. Bir örnek ile bu sistemin çalışma şekli bir örnek ile aşağıda gösterilmiştir [27].

Anahtar Üretimi:

- Şifreleme sisteminde önce kullanıcı anahtarları oluşturulur.
  - İki asal sayı: $p=71, q=83$
  - İki asal sayının(p ve q) çarpımından: $N= 71 \times 83=5893$

- Φ (n) =(p-1)*(q-1) = 5740

  OBEB (Φ (n) , e) =1 ve 1< e < Φ (n) koşullarını sağlayan *e* seçilir. $e = 33$

- $d \equiv e^{-1} \bmod \Phi(n)$ denkliğinden 1< d < Φ (n) ve OBEB(Φ (n), d) =1 koşullarını sağlayan özel anahtar *d* belirlenir. *d = 2957*

ü *(e=33, n=5893)* açık anahtar olarak yayınlanır.

Şifreleme:

ü Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metni *M=1320* şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

$C = M^e \bmod N$

$C = 1320^{33} \bmod 5893 = 2143$

Şifre açma:

ü Şifreli mesaj standart RSA'dan farklı olarak çözülür.

- Öncelikle d_p ve d_q değerleri hesaplanır.

  $d_p = d \bmod (p-1)$ è $d_p = 2957 \bmod 70 = 17$ *(4.2)*

  $d_q = d \bmod (q-1)$ è $d_q = 2957 \bmod 82 = 5$ *(4.3)*

- Sonra M_p ve M_q değerleri hesaplanır.

  $M_p = C^{dp} \bmod p$ è $M_p = 2143^{17} \bmod 71 = 42$ *(4.4)*

  $M_q = C^{dq} \bmod q$ è $M_q = 2143^{5} \bmod 83 = 75$ *(4.5)*

- Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

  $M = M_p + p.v, \quad v = (M_q - M_p) p^{-1} \bmod q$ *(2.8)*

  Euclidean algoritması ile $p^{-1} \bmod q$ è $71^{-1} \bmod 83 = 76$

  $v = (75-42)76 \bmod 83 = 18$

- $M = 42 + 71*18 = 1320$

CRT-RSA'nın Büyük-O analizi

Şifre açma işlemini irdelersek: CRT RSA sisteminde büyük sayılarla yapılan $M = C^d \bmod N$ işlemi yerine $M_p = C^{dp} \bmod p$, $M_q = C^{dq} \bmod q$ işlemleri yapılmaktadır.

Normal RSA'nın $O(n^3)$ olduğunu kısım 3.3 de gösterilmiştir. CRT RSA sisteminde ise *N* sayısı *n bit* iken *p, q, d_p ve d_q* değerleri *n/2* bittir. Buna göre $M_p = C^{dp} \bmod p$ değerinin hesaplanmasında *n/2 bitlik* sayıların çarpma işlemi *n/2 bit* kadar tekrarlanacaktır. Çarpma işlemi O(n/2*n/2) dir ve bu işlem *n/2 bit* kadar tekrarlanınca Büyük-O analizinin sonucu $O((n/2)^3)$ olur.

Şifre açmanın $M_q = C^{dq} \bmod q$ kısmıda $O((n/2)^3)$ dir. Bu iki kısımı birleştirince CRT RSA nın Büyük-O analiz sonucu $O(2(n/2)^3)$dir.

4.1.2.2. Rebalanced-CRT RSA

Kanadalı bilim adamı M.J. Wiener 1990 da yeni bir RSA sisteminden söz etti. Özel anahtarın öncelikle belirlenmesini öngören bu sistemde şifre açma kısmında kullanılacak olan d_p ve d_q değerlerinin daha önceden belirlenen bir bit uzunluğunda olması sağlanarak sistem hızlandırılmıştır [25,26,28].

Rebalanced-CRT RSA'nın çalışma sistemi:

Bu sistemde diğer sistemlerden farklı olarak önce özel anahtar üretilir.

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - *n/2 bit* büyüklüğünde iki adet asal sayı *p ve q* belirlenir.
  - Bu asal sayıların çarpımından *n bit* büyüklüğünde *N* sayısı hesaplanır. *N=p.q*
  - Daha sonra OBEB(d_p , *p-1*) ve OBEB(d_q , *q-1*) koşullarını sağlayan, *s-bit* büyüklüğünde d_p ve d_q değerleri belirlenir.
  - Belirlenen bu d_p ve d_q değerlerinden Garner'ın Çin Kalan Teoremi kullanılarak özel anahtar *d* hesaplanır.

$$\bar{d} \equiv (d_q - d_p)\, p^{-1} \bmod q \qquad (4.6)$$

$$d = d_q + \bar{d}.q \qquad (4.7)$$

  - $d.e \equiv 1 \bmod \Phi(n)$ denkliğinden açık anahtar *e* hesaplanır.
- Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

$$C = M^e \bmod N$$

Şifre açma:

- Şifreli mesaj klasik CRT RSA'daki gibi çözülür.
  - M_p ve M_q değerleri hesaplanır.

$$M_p = C^{dp} \bmod p\,, \quad M_q = C^{dq} \bmod q$$

  - Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

$$M = M_p + p.v\,, \qquad v = (M_q - M_p)\,p^{-1} \bmod q \qquad (2.8)$$

Rebalanced-CRT RSA'nın Büyük-O analizi

Bu sistemde de CRT RSA da olduğu gibi büyük sayılarla yapılan $M = C^d \bmod N$ değerine ulaşmak için $M_p = C^{dp} \bmod p$ ve $M_q = C^{dq} \bmod q$ işlemlerinin değeri hesaplanmaktadır.

Rebalanced-CRT RSA sisteminde *N* sayısı *n bit* iken *p ve q n/2 bittir*. CRT RSA'dan farklı olarak d_p ve d_q değerlerinin her biri *s bit* olarak seçilmiştir. Buna göre $M_p = C^{dp} \bmod p$ değerinin hesaplanmasında *n/2 bitlik* sayıların çarpma işlemi *s bit* kadar tekrarlanacaktır. Çarpma işlemi O(n/2*n/2) dir ve bu işlem *s bit* kadar tekrarlanınca Büyük-O analizinin sonucu $O(s.(n/2)^2)$ olur.

Şifre açmak için $M_q = C^{dq} \bmod q$ işleminde gerçekleşecektir.. Bu yüzden toplam şifre açma işlemi $O(2.s.(n/2)^2)$ dir.

4.1.2.3. Hızlı şifre açma RSA-1

Bu sistemde N sayısını oluşturan iki asal sayının farkı kullanılarak şifre açma işlemi gerçekleştirilir [29].

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - $p < q$ koşulu sağlayan *n/2 bit* büyüklüğünde iki adet asal sayı, *p ve q,* belirlenir.
  - $p + r = q$ ve *k* bir tek sayı iken

    $d \equiv (kq + r) \bmod \Phi(\mathrm{n})$ denkleminden özel anahtar *d* belirlenir.
  - $d \equiv e^{-1} \bmod \Phi(\mathrm{n})$ denkliğinden açık anahtar e hesaplanır.
  - Şifre açma da kullanılacak d_p, d_q değerleri hesaplanır.

$$\begin{aligned} d_p &\equiv d \bmod (p-1) \\ &\equiv kq + r \bmod (p-1) \\ &\equiv k(p+r) + r \bmod (p-1) \\ &\equiv k(r+1) + r \bmod (p-1) \quad (4.9) \end{aligned}$$

$$\begin{aligned} d_q &\equiv d \bmod (q-1) \\ &\equiv kq + r \bmod (q-1) \\ &\equiv k + r \bmod (q-1) \quad (4.8) \end{aligned}$$

Örneğin k=15 iken

$$d_p = 16r + 15 \bmod (p-1) \quad (4.10)$$

$$d_q = r + 15 \bmod (q-1) \quad (4.11)$$

- Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

Şifre açma:

- Şifreli mesaj klasik CRT RSA'daki gibi çözülür.
  - M_p ve M_q değerleri hesaplanır.

    $M_p = C^{dp} \bmod p, \quad M_q = C^{dq} \bmod q$
  - Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

    $M = M_p + p.v, \quad v = (M_q - M_p) p^{-1} \bmod q \quad (2.8)$

Hızlı şifre açma RSA-1'nın Büyük-O analizi

Bu sistemdeki şifre açma işlemi CRT RSA ile aynıdır. Bu sistemin CRT RSA'dan daha hızlı olmasına sebep olabilecek şey d_p ve d_q değerlerinin CRT RSA'dakinden daha küçük olabilme ihtimalidir. d_q ve d_p anahtarlarının büyüklüğünde k sayısı ile N sayısını oluşturan iki asal sayının farkı (r) belirleyicidir. k ve r sayıları kontrollü bir şekilde kullanılıp d_q ve d_p değerlerinin s *bit* büyüklüğünde olduğunu varsayarsak bu sistemin Büyük-O değeri Rebalanced-CRT RSA ile aynı olur: $O(2.s.(n/2)^2)$. N sayısını oluşturan iki asal sayının (p ve q) bir birlerine yakın değerde olmasının getireceği güvenlik sorunundan kurtulmak için r değeri büyük seçilmelidir. Bu durum d_q ve d_p değerlerinin Rebalanced CRT RSA'dakinden büyük olmasına ve dolayısıyla şifre açmanın daha yavaş çalışmasına neden olabilir.

4.1.2.4. Hızlı şifre açma RSA-2

Bir önceki sistemde olduğu gibi bu sistemde de N sayısını oluşturan iki asal sayının farkı kullanılarak şifre açma işlemi gerçekleştirilir [29].

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - $p < q$ koşulu sağlayan *n/2 bit* büyüklüğünde iki adet asal sayı, *p ve q,* belirlenir.
  - $p + r = q$ iken

$$p^i \equiv 1 \bmod (q-1) \qquad q^j \equiv 1 \bmod (p-1)$$

$$p^{i+2} \equiv p^2 \bmod (q-1) \qquad p^{j+1} \equiv q \bmod (p-1)$$

$$\equiv (1-r)^2 \bmod (q-1) \qquad \equiv p + r \bmod (p-1)$$

$$\equiv (r-1)^2 \bmod (q-1) \quad (4.12) \qquad \equiv r + 1 \bmod (p-1) \quad (4.13)$$

Bu denklemler göz önüne alınarak $d \equiv p^{i+2}.q^{j+1} \bmod \Phi(n)$ denkleminden özel anahtar d belirlenir.

  - $d \equiv e^{-1} \bmod \Phi(n)$ denkliğinden açık anahtar e hesaplanır.
  - Şifre açma da kullanılacak d_p, d_q değerleri hesaplanır.

$$d_p \equiv p^{i+2} \bmod(q-1) \equiv (r-1)^2 \bmod(q-1) \quad (4.14)$$

$$d_q \equiv q^{j+1} \bmod(p-1) \equiv r+1 \bmod(p-1) \quad (4.15)$$

- Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

Şifre açma:

- Şifreli mesaj klasik CRT RSA'daki gibi çözülür.
  - M_p ve M_q değerleri hesaplanır.

    $M_p = C^{dp} \bmod p \quad , \quad M_q = C^{dq} \bmod q$

  - Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

    $M = M_p + p.v, \qquad v = (M_q - M_p)\, p^{-1} \bmod q \quad (2.8)$

Hızlı şifre açma RSA-2'nın Büyük-O analizi

Bu sistemin şifre açma işleminde kullanılan d_p ve d_q değerlerinin büyüklüğü sadece iki asal sayının farkına dayanmaktadır. Dolayısıyla d_q ve d_p değerlerinin büyüklüğünü kontrol etmenin daha kolay olacağı ileri sürülebilir. Bu sistemin hızı ise yine d_q ve d_p değerlerinin büyüklüğü ile ilgilidir. Eğer d_q ve d_p değerlerinin *s bit* büyüklüğünde olduğunu varsayarsak bu sistemin zaman karmaşıklığı da $O(2.s.(n/2)^2)$'dir. *s* değerinin büyüklüğü sistemin hızında belirleyicidir.

4.1.2.5. Hızlı şifre açma RSA-3

Bu sistemde de bir önceki iki sistemin bir değişik yöntemidir. Daha önceki iki sistemde olduğu gibi N sayısını oluşturan iki asal sayının farkı kullanılarak şifre açma işlemi gerçekleştirilir [29].

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - $p < q$ koşulu sağlayan *n/2 bit* büyüklüğünde iki adet asal sayı, *p ve q,* belirlenir.
  - $d \equiv p^2.q \bmod \Phi(\mathrm{n})$ denkleminden özel anahtar *d* belirlenir.
  - $d \equiv e^{-1} \bmod \Phi(\mathrm{n})$ denkliğinden açık anahtar *e* hesaplanır.
  - $p + r = q$ iken şifre açma da kullanılacak d_p , d_q değerleri hesaplanır.

$$\begin{aligned} d_p &\equiv d \bmod (p-1) \\ &\equiv p^2.q \bmod (p-1) \\ &\equiv q \bmod (p-1) \\ &\equiv (r+p) \bmod (p-1) \\ &\equiv (r+1) \bmod (p-1) \end{aligned} \quad (4.16)$$

$$\begin{aligned} d_q &\equiv d \bmod (q-1) \\ &\equiv (p^2.q) \bmod (q-1) \\ &\equiv p^2 \bmod (q-1) \\ &\equiv (q-r)^2 \bmod (q-1) \\ &\equiv (r-1)^2 \bmod (q-1) \end{aligned} \quad (4.17)$$

- Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

Şifre açma:

- Şifreli mesaj klasik CRT RSA'daki gibi çözülür.
  - M_p ve M_q değerleri hesaplanır.

$$M_p = C^{dp} \bmod p\,, \quad M_q = C^{dq} \bmod q$$

  - Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

$$M = M_p + p.v\,, \qquad v = (M_q - M_p)\,p^{-1} \bmod q \qquad (2.8)$$

Hızlı şifre açma RSA-3'nın Büyük-O analizi

Bu sistemin şifre açma işlemi, zaman karmaşıklığı Hızlı şifre açma RSA-2 ile aynıdır: $O(2.s.(n/2)^2)$'dir. Bu sistemde de *s* değerinin büyüklüğü sistemin

hızında belirleyici unsurdur. Bu sistem için; anahtar üretim sürecinde daha zorlu işlemler uygulandığından anahtar üretiminin daha yorucu olacağı söylenebilir.

4.1.2.5. Çok asallı RSA (MultiPrime RSA)

1998 de geliştirilmiş olan bu sistemde iki asal sayı yerine *k* adet asal sayı $p_1, p_2, p_3.....p_k$ kullanılmaktadır [21,26].

Çok asallı RSA'nın çalışma sistemi:

Bu sistem *N* değerini birçok asal sayı kullanarak büyütmeyi uygun görür.

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - *n/k bit* büyüklüğünde k adet asal sayı $p_1, p_2, p_3.....p_k$ belirlenir.
  - Bu asal sayıların çarpımından n bit büyüklüğünde *N* sayısı hesaplanır. $N = \prod_{i=1}^{k} p_i$
  - $\Phi(N) = \prod_{i-1}^{k} (p_i - 1)$ çarpımı ile ortak böleni 1 olan açık anahtar "e" sayısı seçilir.
    $1< e < \Phi (n)$
  - $d \equiv e^{-1} \bmod \Phi(n)$ denkliğinden özel anahtar *d* hesaplanır.
    $1< d < \Phi (n)$
- Bu değerlerden ***(e,n)*** açık anahtar olarak yayınlanır.

Şifreleme:

- Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür
- Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

$$C = M^e \bmod N$$

Şifre açma:

- Şifreli mesaj klasik CRT RSA'daki gibi çözülür.
  - Öncelikle d_i değerleri hesaplanır.

    $d_i = d \bmod (p_i - 1)$

- Sonra M_i değerleri hesaplanır.

  $M_i = C^{di} \bmod p_i$

- Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

  $M = M_p + p.v \quad , \qquad v = (M_q - M_p)\, p^{-1} \bmod q$ *(2.8)*

Çok Asallı RSA'nın Büyük-O analiz

Bu sistemde büyük sayılarla yapılan $M = C^d \bmod N$ değerine ulaşmak için *k* adet $M_i = C^{di} \bmod p_i$ işlemi yapılmaktadır.

N sayısı *n bit* iken $p_1, p_2 .... p_k$ ve $d_1, d_2 .... d_k$ değerlerinin her biri *n/k* bittir. Buna göre $M_i = C^{di} \bmod p_i$ değerinin hesaplanmasında *n/k bitlik* sayıların çarpma işlemi *n/k bit* kadar tekrarlanacaktır. Çarpma işlemi O(n/k*n/k) dir ve bu işlem *n/k bit* kadar tekrarlanınca Büyük-O analizinin sonucu $O((n/k)^3)$ olur.

Şifre açmak için $M_i = C^{di} \bmod p_i$ işleminden ***k adet*** yapıyoruz. Bu yüzden toplam şifre açma işlemi $O(k.(n/k)^3)$ dir.

4.1.2.6. Kuvvet RSA

Rsa'nın bu çeşidinde *N* sayısını oluşturan *p ve q* değerlerinden birinin ikinci kuvveti alınır [30,31].

Kucvvet RSA'nın çalışma sistemi:

Bu sistemden diğer sistemlerden farklı kuvvet alınarak *N* sayısının değerini büyütülür.

Anahtar üretimi:

ü Önce kullanıcı anahtarları oluşturulur.

- *n/3 bit* büyüklüğünde iki adet asal sayı *p ve q* belirlenir.
- Bu asal sayıların birinin karesi alınır ve diğeri ile çarpılarak *n bit* büyüklüğünde *N* sayısı hesaplanır. $N=p^2.q$
- Φ (n) değeri ile ortak böleni bir olan açık anahtar *e* sayısı seçilir.
  $1< e < \Phi (n)$

- $d.e \equiv 1 \bmod \Phi(n)$ denkliğinden açık anahtar e hesaplanır.

ü Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

ü Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür

ü Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

$C = M^e \bmod N$

Şifre açma:

ü Şifreli mesaj klasik CRT RSA'daki gibi çözülür.

- Öncelikle d_p ve d_q değerleri hesaplanır.

  $d_p = d \bmod (p-1)$; $d_q = d \bmod (q-1)$

- Sonra M_p ve M_q değerleri hesaplanır.

  $M_p = C^{dp} \bmod p$; $M_q = C^{dq} \bmod q$

- M_{p^2} 'yi hesaplaya bilmek için *Hensel Lifting* Teoremi Kullanılır.

  $M_{p2} = M_0 + (C - M_0^{\,e})(e^{-1} \bmod p)(C^{dp-1} \bmod p) \bmod p^2$ *(2.9)*

- Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

  $M = M_{p2} + p^2.v$, $v = (M_q - M_{p2})(p^2)^{-1} \bmod q$ *(2.8)*

Kuvvet RSA'nın Büyük-O analizi

Bu sistemde büyük sayılarla yapılan $M = C^d \bmod N$ değerine ulaşmak için M_{p2} ve $M_q = C^{dq} \bmod q$ işlemlerinin değerleri hesaplanmaktadır.

N sayısı *n bit* iken *p, q,* d_p ve d_q değerleri *n/3 bittir*. Buna göre $M_q = C^{dq} \bmod q$ değerinin hesaplanmasında *n/3 bitlik* sayıların çarpma işlemi *n/3 bit* kadar tekrarlanacaktır. Çarpma işlemi O(n/3*n/3) dir ve bu işlem *s bit* kadar tekrarlanınca Büyük-O analizinin sonucu $O((n/3)^3)$ olur.

Şifre açmak için M_{p2} işlemi gerçekleştirilirken *Hensel Lifting* teoremi kullanılır. Hensel Lifting işleminde kullanılacak olan $e^{-1} \bmod p$ anahtar üretimi sırasında hesaplanıp kullanılacağından bir yük getirmeyecektir. $C^{dp-1} \bmod p$ değeri M_0 hesaplanırken tutulabilir. Bu kısımda $M_0{}^{e} \bmod p^2$ işlemi yük tutacaktır. p^2 , *2n/3 bit* iken bu işlem $O(4n^3/27)$ dir. Kuvvet RSA 'nın toplam değeri ise $O(5n^3/27)$ olur.

Bu sistemde belirtilmesi gereken bir noktada anahtar üretim sürecinin standart RSA'ya göre daha zor olduğudur. Şifre açma kısmını hızlandırabilmek için $e^{-1} \bmod p$, $(p^2)^{-1} \bmod q$ değerleri de anahtar üretim aşamasında gerçekleşeceği için bu kısmın üretim süresi daha uzundur. Fakat bu değerler sadece bir kere hesaplanacağı için bu durumun getireceği yük göz ardı edilebilir.

4.2. Önerilen RSA Algoritması

Bu bölümde geniş anahtar havuzu sağlayan fakat şifre açma kısmını verimsizleştirip, h^2 kat daha yavaş çalışmasına neden olan Verimli RSA Şifreleme Şemasının bu verimsizliğini ortadan kaldırabilmek, bu şemayı hızlandırabilmek için Kuvvet RSA, Rebalanced-CRT RSA sistemlerinin özelliklerinden faydalanarak diğer sistemlerden farklı çalışan bir RSA algoritması önerilmiştir. Uygulaması yapılarak standart RSA ile karşılaştırılmıştır. Sonuçları irdelenmiştir. Bu algoritmanın çalışma şekli şöyledir:

Anahtar üretimi:

- Önce kullanıcı anahtarları oluşturulur.
  - *n/3 bit* büyüklüğünde iki adet asal sayı *p ve q* belirlenir.
  - Bu asal sayıların çarpımından n bit büyüklüğünde *N* sayısı hesaplanır. $N=p^2.q$
  - Sonra bir *h* sayısı belirlenir. Doğrusal cebir yaklaşımı ile *g* sayısı oluşturulur.

    $$g=(p^h-1)(p^h-p)(p^h-p^2)....(p^h-p^{h-1}).(q^h-1)(q^h-q)(q^h-q^2)....(q^h-q^{h-1}) \quad (3.10)$$
  - Rebalanced-CRT sisteminde olduğu gibi önce özel anahtar hesaplanır.

$$d \equiv \left.\begin{matrix} d_p \bmod (p-1) \\ d_q \bmod (q-1) \end{matrix}\right\} \quad \text{ve} \quad \begin{matrix} \text{OBEB}(d_p, p-1) = 1 \\ \text{OBEB}(d_q, q-1) = 1 \end{matrix}$$

Koşullarını sağlayan *d, dp ve dq* özel anahtarları bulunur.

- $e \equiv d^{-1} \bmod g$ denkliğinden açık anahtar e belirlenir.

ü Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

ü Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür

ü Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin *M* bloklarını şifreler ve elde etiği şifreli değerleri *C* karşıya gönderir.

$$C = M^e \bmod N$$

Şifre açma:

ü Şifreli mesaj klasik CRT RSA'daki gibi çözülür.

§ Sonra M_p ve M_q değerleri hesaplanır.

$$M_p = C^{dp} \bmod p \; ; \; M_q = C^{dq} \bmod q$$

§ M_{p^2} 'yi hesaplaya bilmek için *Hensel Lifting* Teoremi (2.9) kullanılır.

§ M_{p^2} , M_q değerlerinden (2.8) deki Çinli Kalan Teoremi ile *M* değerine, açık metne ulaşılır.

*Örnek***:**

Önerilen RSA sisteminin bir örnek ile gösterilirse:

Anahtar üretimi:

ü Önce kullanıcı anahtarları oluşturulur.

- *n/3 bit* büyüklüğünde iki adet asal sayı *p=1343491 ve q =1543489*
- Bu asal sayıların çarpımından *n bit* büyüklüğünde *N* sayısı hesaplanır.

  $N=p^2.q = 1343491^2 * 1543489 = 2785948356890785609$

- Sonra bir *h* sayısı belirlenir ve bu *h* sayısı kullanılarak doğrusal cebir yaklaşımı ile *g* sayısı oluşturulur.

  $g=(p^h-1)(p^h-p)(p^h-p^2)....(p^h-p^{h-1}).(q^h-1)(q^h-q)(q^h-q^2)....(q^h-q^{h-1})$

  h=2 **è** g = $(p^2-1).(p^2-p).(q^2-1).(q^2-1)$

  g= 184906677970121924999004306396373874675785236480000

- $d \equiv \left.\begin{matrix} d_p \bmod (p-1) \\ d_q \bmod (q-1) \end{matrix}\right\}$ ve $\begin{matrix} OBEB(d_p, p-1) = 1 \\ OBEB(d_q, q-1) = 1 \end{matrix}$

  Koşullarını sağlarından *d=611899; d_p =1955389 ; d_q =2155387*

  § $e \equiv d^{-1} \bmod g$ denkliğinden açık anahtar

  e=302184965116991407076991965007907963039300 99.

ü Bu değerlerden *(e,n)* açık anahtar olarak yayınlanır.

Şifreleme:

ü Şifrelenecek olan açık metni öncelikle [0, n-1] arasındaki pozitif tamsayı bloklar haline dönüştürülür

ü Mesaj gönderecek kişi; alıcının açık anahtarını kullanarak açık metnin bloklarını şifreler *M=13476843216237* ve elde etiği şifreli değerleri *C* karşıya gönderir.

$C = M^e \bmod N$ = 2187980553308224401

Şifre açma:

ü Şifreli mesaj klasik CRT RSA'daki gibi çözülür.

§ d_p =1955389 ; d_q =2155387 değerleri biliniyor

$M_0 = M_p = C^{dp} \bmod p$ =175145; $M_q = C^{dq} \bmod q$ =209302;

§ M_{p^2} 'yi hesaplaya bilmek için *Hensel Lifting* Teoremi Kullanılır.

$M_{p2} = M_0 + (C - M_0^{\,e})(e^{-1} \bmod p)(C^{dp-1} \bmod p) \bmod p^2$ *(2.9)*

=175145+ 332328620742 . 611899 . 492782

=842066746670

§ Çin Kalan Teoremi kullanılarak açık metne ulaşılır.

$M = M_{p2} + p^2.v \qquad v = (M_q - M_{p2})(p^2)^{-1} \bmod q$

v=(209302 – 842066746670) * (1046042) mod q =7

M=842066746670 + 1804968067081 * 7

= 13476843216237

Önerilen RSA'nın Büyük-O analizi

Bu sistemde önce *d* özel anahtarını belirlediğimizden ve *g* değerinin çok büyük olduğundan açık anahtarımızın *(e)* değeri oldukça yüksek oluyor. Açık anahtar olarak *N* sayısından, *n bitten* daha büyük bir değer elde ediyoruz. Bu durum şifreleme kısmına zaman bakımından yük getireceğinden bu tezdeki uygulama sırasında modüler aritmetiğin bir özelliğinden yararlanılmaya çalışılmıştır(1). Modüler aritmetikte *mod N* değeri en fazla *N* kadar çeşitlilik gösterebilir. *e* sayımız *N* değerinden büyük olduğundan bu değerlerin birbirlerini tekrarladığı anlamına gelir. Değerlerin tekrarlama anından itibaren sitem durdurulursa mod işlemi en kötü ihtimalle *N* kadar yapılmış olur. *(Örnek : 27^{521} mod 89 = 27^{81} mod 89 = 76)* Bu sayede şifreleme kısmının yük getirmesinin önüne geçilmeye çalışılmıştır.

Şifre açma işlemi incelenirse Kuvvet CRT RSA da olduğu gibi $M = C^d \bmod N$ değerine ulaşmak için M_{p2} ve $M_q = C^{dq} \bmod q$ değerleri hesaplanmalıdır.

Kuvvet RSA daki Büyük-O ($5n^3/27$) değeri bu sistem için de geçerlidir. Yine bu sistemde Kuvvet RSA'da olduğu gibi şifre açma kısmını hızlandırabilmek için anahtar üretim kısmında $e^{-1} \bmod p$, $(p^2)^{-1} \bmod q$ değerlerinin hesaplanması gerekmektedir. Bu değerlerin yanında *g* değerinin hesaplanması, ayrıca açık anahtarın *mod g* 'ye göre belirlenmesi anahtar üretim sürecini yavaşlatmaktadır. Fakat bu değerler sadece bir kere hesaplanacağı için bu durumun getireceği yük göz ardı edilebilir

(1) Uygulamada kullanılan yöntemin detayları sonuçları, olumlu ve olumsuz yanları kısım 3.6'da anlatılmıştır.

4.3. RSA Uygulamasının Sonuçları ve Değerlendirilmesi

Bu tezde şifre açma kısmı Standart RSA'dan h^2 kat kadar daha yavaş olan Verimli RSA Şifreleme Şemasını hızlandırmak için bazı şifre açma algoritmaları incelenmiş ve yeni bir algoritma önerilmiştir. Bu kısımda ise Önerilen RSA algoritması; Verimli Şifreleme Şeması, verimli şemadan h^2 *kat* daha hızlı çalışan Standart RSA ve Hızlı Şifre Açma RSA-1 ile hız bakımından karşılaştırılmıştır. Bu uygulamada Hızlı Şifre Açma RSA-1'ın hızında etken olan r değeri iki farklı aralıkta alınarak aynı N değeri için iki kere çalıştırılmıştır. Bu uygulamanın birinde r değeri düşük tutulurken diğerinde daha büyük tutulmuştur. Bu algoritmalar java programı ile kodlanmış ve birbirine yakın N sayıları ele alınarak değerleri hesaplanmıştır.(2)

Uygulama sonucunda bu algoritmaların şifre açma kısmında harcadığı zamanın milisaniye cinsinden değerleri Çizelge 4.1'de gösterilmektedir. Bu çizelgede görüldüğü üzere; Önerilen RSA'nın, Verimli RSA Şifreleme Şeması ve Standart RSA karşı üstünlüğü ortadadır. r sayısının küçük tutulduğu durumda Hızlı Şifre Açma RSA-1 daha hızlı iken r sayısının değeri artıkça hız Önerilen RSA'ya yaklaşmaktadır.

Hızlı Şifre Açma RSA-1 sisteminde asallar arası farkın (r) küçük olması sistemi güvensiz kılacağı için bu değerin küçük seçilmemesi önemlidir. Asal sayı farkının $r < n^{1/4}$ olduğu durumlarda sistem güvensizdir. Bu durumda Fermat'ın çarpanlara ayırma tekniği sistemin anahtarlarını tehdit etmektedir. Daha güvenli bir sitem için asallar arası farkın yüksek olması gerekmektedir. Örneğin Fermat tekniği ile yapılan ataklara karşı koyabilmek için N sayısı *1024 bit* iken asallar arası fark en az $r \approx n^{412}$ olmalıdır[29]. Yüksek seçilen r değeri de sistemi yavaşlatmaktadır. r değeri artıkça Hızlı Şifre Açma RSA-1 sisteminin hız değerlerinin Önerilen RSA'nın değerlerine yaklaştığı görülmektedir. Daha yüksek güvenlik için Hızlı Şidre Açma RSA-1'in daha yavaş çalışacağı öngörülmektedir.

---

(2)Karşılaştırmanın yapıldığı bu çalışmada bilgisayarın işletim gücü önemli değildir. Önemli olan aynı işletim gücüne sahip bilgisayarlarda iki algoritmanın gösterdiği farklılıktır.

Çizelge 4.1. Algoritmaların şifre açma kısımlarının karşılaştırılması

ŞİFRE AÇMA (DECRYPTION) (milisaniye)					
N	**Verimli RSA Şifreleme Şeması**	**Hızlı Şifre Açma RSA-1[(3)]**	**Hızlı Şifre Açma RSA-1[(4)]**	**Standart RSA**	**Önerilen RSA[(5)]**
N ≅ 760 000	**50879531**	**16**	**93**	**11874**	**112**
N ≅ 9 900 000	**125331244**	**98**	**234**	**24262**	**287**
N ≅ 65 000 000	**677612136**	**445**	**985**	**72413**	**1145**
N ≅ 530 000 000	**356987655217**	**3802**	**13547**	**7521231**	**12374**
N ≅ 2 200 000 000	**8511125647145**	**5622**	**20544**	**18673217**	**24268**
N≅ 10 450 000 000	**…**	**7895**	**31598**	**45177328**	**29367**

Algoritmanın şifreleme kısmında harcadıkları zamanın değerleri ise Çizelge 4.2'de gösterilmektedir. Kısım 4.5.'de "Önerilen RSA" sisteminin şifreleme kısmı için anahtar havuzu konusunda fayda sağladığı anlatılmış, şifreleme kısmına zaman bakımından yük getirmemesi için modüler aritmetikten yararlanılmaya çalışılacağı ifade edilmiştir. Uygulama sonucunda çıkarılan çizelgeye bakıldığında "Önerilen RSA" sisteminin zaman bakımından yük getirmek bir yana avantaj sağladığı görülmektedir.

Önerilen sistemde açık anahtarın çok büyümesi, hatta N değerini aşmasından dolayı şifreleme kısmının zaman bakımından verimsizleşmesi beklenmektedir. Kısım 4.5. de modüler aritmetiğin özelliği ile bu yükten kurtulmaya çalışılacağı belirtilmiş ve bir örnek verilmiştir. Eğer benzer bir örnek ele alınırsa aşağıdaki çizelgede ortaya koyulan sonuçlar daha iyi anlaşılabilir.

[(3)]Hızlı Şifre Açma RSA-1'de şifre açma kısmının hızını belirleyen r değeri (p ve q arasındaki fark) 0,06.N% civarında seçilmiştir. Düşük r değerinin güvenlik sorunu getireceği unutulmamalıdır.

[(4)] Hızlı Şifre Açma RSA-1'de şifre açma kısmının hızını belirleyen r değeri (p ve q arasındaki fark) 0,12.N% civarında seçilmiştir.

[(5)]Önerilen RSA'nın şifre açma kısmında kullanılan $C^{dp-1} \bmod p$ değeri M_0 hesaplanırken tutulmamış tekrardan hesaplanmıştır. Buna rağmen sistemin verimliliği ortadır.

Çizelge 4.2. Algoritmaların şifreleme kısımlarının karşılaştırılması

ŞİFRELEME (ENCRYPTION) (milisaniye)				
N	**Verimli RSA Şifreleme Şeması**	**Hızlı Şifre Açma RSA-1**	**Standart RSA**	**Önerilen RSA[(6)]**
N ≅ 760 000	**65458789**	**5789**	**6272**	**223**
N ≅ 9 900 000	**148231481**	**45087**	**42362**	**5232**
N ≅ 65 000 000	**581120632**	**65124**	**62813**	**14563**
N ≅ 530 000 000	**274061241244**	**4321454**	**4837621**	**1689461**
N ≅ 2 200 000 000	**5598641287111**	**32504789**	**34321937**	**12400657**
N≅ 10 450 000 000	**…**	**69874556**	**67609943**	**24177328**

Örneğin $18^{15487} \bmod 221$ değerini hesaplamak için 18'in kuvvetlerinin mod 221'e göre değerleri bulunur.

$18^1 \bmod 221 = 18$

$18^2 \bmod 221 = 103$

$18^3 \bmod 221 = 86$

$18^4 \bmod 221 = 1$

"1" kalanı bulunduğunda durulur. Çünkü "1" değerinden sonra mod değerleri birbirlerini tekrarlayacaktır. "1" değerini sağlayan sağlayan kuvvet bir sonraki adımda mod değeri olarak kullanılır.

$15487 \bmod 4 = 3$

Bulunan sonuç; ilk kuvvet değeri ile eş sonucu veren yeni kuvvettir.

$18^{15487} \bmod 221 = 18^3 \bmod 221 = 86$

Yukarıda görüldüğü üzere 15487 kere mod işlemi yapmak yerine sadece 6 işlem ile sonuca ulaşılmıştır. İşte bu sebepten dolayı şifreleme kısmı Standart RSA'ya göre daha hızlı olmuştur.

[(6)]Önerilen RSA'nın şifreleme kısmında mod değerleri tutulmamış her seferinde tekrardan hesaplanmıştır.

Fakat hemen belirtmek gerekir ki; bu çalışma şekli tam olarak istediğimiz bir çalışma şekli olmayabilir. Zira 221 sayısını oluşturan 13 ve 17 değerlerinin katları mod işlemine girdiğinde çıkan mod değerleri içinde 1 sayısı olmaz. Mod değerleri tekrara herhangi bir sayıdan sonra başlar. Bu durumda devreye güvenilirlik prensibi girer. Sistemin kendisinden beklenen şeyi eksiksiz ve fazlasız olarak her çalıştırıldığında tutarlı bir şekilde yapması prensibi sekteye uğrar. Bu durumu düzeltmek için iki yol uygulanabilir:

- Mod alma işlemi yapılırken bulunan her değer sıraya dizilerek veya karşılaştırılarak, tekrar yakalandığında durulabilir. Böyle bir algoritmada sıraya koyma işlemi sitemi biraz daha yavaşlatacaktır. Fakat hangi oranda yavaşlatacağı soru işaretidir. Bazı mod işlemleri çok hızlı olabilecekken bazı işlemler ise Standart RSA'ya yaklaşabilir hatta geçebilir.

- Diğer bir yol olarak anahtarların belirlenmesi sırasında e anahtarının bit uzunluğunun en fazla n olacak şekilde küçük olması sağlanmalıdır. Bu durum şifreleme kısmını Standart RSA, Hızlı Şifre açma RSA-1 ve diğer tüm algoritmalar ile birebir aynı hale getirecektir. Ne fazladan bir yük ne de bir kolaylık sağlayacaktır. Çünkü şifre açma kısmı için d sayısının hatta yukarıdaki modüler aritmetik göz önüne alınırsa d_p *ve* d_q sayılarının büyüklüğü önemli değildir.[7] Önemli olan p *ve* q sayılarının büyüklüğüdür. Bu sayılarda da değişmediğinden dolayı şifre açma kısmı aynı şekilde çalışmaya devam edecektir.

İkinci yol ele alındığında şifre açma tablosu Çizelge 4.1'deki aynı iken şifreleme tablosunda Önerilen RSA'nın değerleri Standart RSA ve Hızlı Şifre açma RSA-1'nınkilerle aynı olacaktır. Fakat anahtar üretim kısmının az da olsa zorlu bir süreçten geçeceği söylenebilir. Bu sistemde seçilmiş bir açık anahtardan özel anahtar üretmek biraz daha zaman alabilir. Fakat bu işlem sadece bir defa yapılacağından göz ardı edilebilir.

---

[7] p asal iken $m^x \bmod p = 1$ koşulunu sağlayan bir x değeri muhakkak vardır 1<x<p

4.4. Önerilen RSA Sisteminin Güvenliği

Bu sistemin güvenliği bu tezde incelenen tüm RSA sistemlerinde olduğu gibi N sayısının çarpanlarına ayrılabilme zorluğuna dayanır. Çarpanlarına ayırma işleminin matematiksel olarak herhangi bir dayanağı olmadığı için N sayısı ne kadar büyük ise sistem o kadar güçlüdür. O yüzden önerilen bu sistem p ve q sayısının elde edilmesi konusunda herhangi bir zafiyet oluşturmamaktadır.

Önerilen bu sistemin seçilmiş metin saldırılarına karşı ise daha güçlü olduğu iddia edilebilir. Anahtar havuzunun genişlemesi seçilmiş metin saldırılarında denenecek olan anahtarların sayının artması ve şifre kırma süresinin uzaması anlamına gelir.

5. SONUÇLAR ve ÖNERİLER

Asimetrik kriptografinin en önemli şifreleme sistemlerinden biri RSA sistemidir. Asimetrik şifrelemenin en önemli sorununu olan hız problemi RSA içinde geçerlidir. Asimetrik algoritmalar dijital imza gibi birçok avantaja sahipken yavaş olmaları en büyük dezavantajdır. Bu yüzden bazı düzenlemelerle RSA'nın şifre açma kısmını hızlandıran yeni RSA sistemleri bu tezde mercek altına alınmıştır.

Standart RSA'da şifre açma işlemi $O(n^3)$ iken Çin Kalan Teoremli RSA ile başlayan gelişmelerde zaman karmaşıklığı ilk olarak $O(2.(n/2)^3)$'e düşmüştür. Bu da mevcut sistemi dört kat hızlandırmak anlamı taşımaktadır. Diğer gelişmelerle Rebalanced CRT-RSA ve Hızlı Şifre Açma RSA sistemleri *2n/s*, Çok Asallı RSA k^2, *Kuvvet* RSA ise *5,4* kat daha hızlandırmaktadır.

RSA üzerinde yapılan çalışmaların çoğu şifre açma kısmını hızlandırmaya yönelik çalışmalardır. Şifreleme kısmına yönelik çalışma fazla mevcut değildir. 2008 yılında şifreleme kısmını geliştirmek için; daha geniş bir anahtar havuzu oluşturan Verimli RSA Şifreleme Şeması ileri sürüldü. Fakat bu sistem şifreleme kısmını verimli yaparken şifre kısmını verimsizleştiriyordu. Bu sistem ile RSA'nın özellikle şifre açma kısmının zaman karmaşıklığı $O(n^3.h^2)$'e kadar yükselmektedir. Bu sistemi hızlandırmak için Kuvvet RSA ve Rebalanced CRT RSA sistemlerinden, modüler aritmetikten yaralanılarak yeni bir RSA algortiması önerilmiştir. Önerilen bu sistemin şifre açma hızı Kuvvet RSA ile aynıdır. Bu da Verimli RSA Şifreleme Şemasının şifre açma kısmının $5,4.h^2$ kat daha hızlanması demektir.

Nesne tabanlı programlama ile Önerilen RSA; Standart RSA, Verimli Şifreleme Şeması, Hızlı Şifre Açma RSA-1 karşılaştırılmıştır. Bu karşılaştırma ile şifre açma ksımının hızlanışı gözler önüne serilirken, şifreleme kısmında da hızlanma geçekleşmiştir. Şifreleme kısmında hızlanmaya sebep olan modüler aritmetiğin bir özelliğidir. Bu özellik beraberinde güvenilirlik prensibine soru işareti getirmiştir. Fakat benzer bir yapılanma ile hızda yavaşlama olsa dahi sorunun giderilebileceği öngörülmüştür. Ya da bu soru işareti şifreleme kısmında hızlanma istemerek, modüler aritmetiği şifreleme kısmından çıkarıp anahtar üretim sürecini az daha yorarak da ortadan kaldırılmıştır. Böylelikle daha geniş

anahtar havuzuna sahip, Standart RSA'dan daha hızlı ve onun kadar güçlü hatta seçilmiş metin saldırılarına karşı daha dirençli bir algortima ortaya çıkarılmıştır.

Bu tezde önerilen sistem birçok çalışmaya zemin hazırlayabilir. Öncelikle şifreleme kısmını hızlandıracağını ön görülen algoritmanın sistemi ortalama ne kadar hızlandıracağı ya da hızlandırıp hızlandırmayacağı incelenebilir. Ya da Rebalanced CRT RSA'daki gibi ***s-bitlik dp, dq*** değerlerinden Çinli Kalan Teoreminin yardımıyla ***d*** anahtarının bulunması için çalışma yapılabilir. Önerilen sistemde anahtar havuzunu genişleten ***g*** sayısı yüzünden bu yaklaşım kullanılamamıştı. Bu yaklaşım bu sisteme oturtulup şifre açmanın hızı bir kademe daha artırılabilir. Eğer Rebalanced CRT RSA'nın şifre açmayı hızlandıran bu özelliği de önerilen sisteme uygulanabilirse sistem ***n/(3.s)*** kadar daha hızlanabilir. Ya da Hızlı Şifre Açma RSA'daki gibi ***r*** değerinin kontrolü ile ***s-bitlik dp, dq*** değerleri yaratılabilirse yine ***n/(3.s)*** kadar hızlanma sağlanabilir. Önerilen RSA'da ***g*** sayısı sebebiyle bu yaklaşım kullanılamamıştı.

Verimli RSA Şifreleme Şemasının yaratıcıları ***h*** değeri ile ***g*** sayısını oluşturmaktadır ve Hill Cipher (matris) sisteminin kullanılması halinde h^2 blok birden gönderilebileceğini ileri sürülmektedirler. Bu da uzun bir mesajın hızlıca, topluca gönderilebileceği müjdesidir. Bu çalışmadaki sistemin matrisler üzerinde ne sonuç vereceği de tartışılabilir.

KAYNAKLAR

[1] Babaoğlu, A., "Kriptolojinin Geçmişi Bir Şifreleme Algoritması Kullanmadan Önce Son Kullanım Tarihine Bakın!", Bilim ve Teknik, Tubitak, 24-27, 2009.

[2] Çeşmeci, M., "Kriptoloji tarihi", UEKA dergisi, 1, 20-31, 2009.

[3] Çimen, C., Akyelek, S. ve Akyıldız, E., *Şifrelerin Matematiği Kriptografi*, Odtü Yayıncılık, Ankara, 2008.

[4] Atunbaş, A., "Şifreli söyleşi", UEKA dergisi, 2, 6-23, 2010.

[5] Süer, S., Tüyeni E. ve Koç, Ç., "Tubitak UEKAE'nin başarı yolculuğu", UEKA Dergisi, 1, 9-19, 2009.

[6] Anonim, *Bilişim Güvenliği*, Pro-G Bilişim Güvenliği ve Araştırma Ltd., 2003.

[7] Tekerek, M., "Bilgi Güvenliği Yönetimi", Kahramanmaraş Sütçü İmam Üniversitesi Fen ve Mühendislik Dergisi, 1, 132, 2008.

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

[9] Yerlikaya, T., Buluş, E. ve Arda, D., "Asimetrik Kripto Sistemler ve Uygulamaları", II. Mühendislik Bilimleri Genç Araştırmacılar Kongresi, 2005.

[10] Anonim, Tubitak Açık Anahtar Altyapısı Eğitim Kitabı, 2010. http://www.kamusm.gov.tr/tr/bilgideposu/belgeler/teknik/aaa/index.html?kriptoanalizyontemleri.html

[11] Eren, M., "Açık Anahtarlı Kriptografi", Penguence Dergisi, 2, 2005.

[12] Sarad A.V. ve Huettenhain, J., *RSA Encryption Algorithm in a Nut Shell*, 2009. http://data.at.preempted.net/INDEX/articles/rs.pdf

[13] Precup, D., *Big Oh*, McGill University, 2008. http://www.cs.mcgill.ca/~dprecup/courses/IntroCS/Lectures/comp250-lecture14.pdf

[14] Stinson, D., *Cryptography Theory and Practice*, Chapman & Hall / CRC, A.B.D., 2006.

[15] Andeson, J. ve Bell, J., *Number Theory with Application*, Prentice Hall, Upper Saddle River, A.B.D.,1997.

[16] Kendirli, B., *Number Theory with Cryptographic Applications*, Mavi Matbaacılık, İstanbul, 2006.

[17] Crandall, R. ve Pomerance, C., *Prime Numbers a Computational Perspective*, Springer, A.B.D., 2001

[18] Delfs, H. ve Knebl, H., *Introduction to Cryptography principles and Application*, Springer, A.B.D., 2002.

[19] Yen, A. ve Kim, D., *Cryptanalysis of Two Protocolls for RSA wiith CRT Based on Faullt Infectiion*, Springer Berlin, Almanya, 2006.

[20] Shoup, V., *Computational Introduction to Number Theory and Algebra*, Cambridge University Press, A.B.D.,2005.

[21] http://www.rsa.com/

[22] Zimmermann, P., "A Proposed Standard Format for RSA Cryptosystems", IEEE Computer Society Press, Los Alamitos, USA, 1986.

[23] Hansen, K., Larsen, T. ve Olsen, K., "On the Efficiency of Fast RSA Variants in Modern Mobile Phones", International Journal of Computer Science and Information Security, No: 3, 2009.

[24] Aboud S., Fayoumi, M., Fayoumi, M. ve Jabbar, H., "An Efficient RSA Public Key Encryption Scheme", Fifth International Conference on Information Technology: New Generations, 2008.

[25] Sun. H. ve Wu,M. *An Approach Towards Rebalanced RSA-CRT with Short Public Exponent*, 2010.
http://eprint.iacr.org/2005/053.pdf

[26] Garg, D. ve Verna, S., "Improvement over PublicKey Cryptographic Algorithm", International Advance Computing Conference, 2009.

[27] Quisquater, J. ve Couvreur, C., "Fast Decipherment Algoritm for RSA Public-Key Cryptosystem", Philips Research Laboratory, Brussels, Belgium, 905-907, 1982.

[28] Ou, H. ve Wei, B., "Multi-factor Rebalanced RSA-CRT Encryption Schemes", Biomedical Engineering and Informatics, 2nd International Conference, 2009.

[29] Penzhom, W., “Fast Decryption Algorithms for the RSA Cryptosystem”, 7th AFRICON Conference, Africa, 2004.

[30] Boneh, D. ve Shacham, H., *CryptoBytes*, RSA Laboratories, 2002.

[31] Kirtane, V. ve Rangan,C., *“Side Channel Attack Resistant Implementation of Multi-Power RSA using Hensel Lifting”,* 2010. http://eprint.iacr.org/2008/368.pdf