Süleyman Demirel Üniversitesi
Fen Bilimleri Enstitüsü Dergisi
Cilt 21, Sayı 2, 631-643, 2017

Süleyman Demirel University
Journal of Natural and Applied Sciences
Volume 21, Issue 2, 631-643, 2017

DOI: 10.19113/sdufbed.70723

RSA Şifreleme Sistemlerinin Kleptografik Arka Kapıları için Güvenlik ve Karmaşıklık Analizi

Emre CERAN *[1], **Mehmet Sabır KİRAZ**[2], **Osmanbey UZUNKOL**[2]

[1]İstanbul Şehir Üniversitesi, Fen Bilimleri Enstitüsü, Bilgi Güvenliği Mühendisliği, Altunizade, İstanbul
[2] TÜBİTAK BİLGEM, Matematik ve Hesaplamalı Bilimler, Gebze, Kocaeli

(Alınış / Received: 17.11.2016, Kabul / Accepted: 07.02.2017, Online Yayınlanma / Published Online: 08.03.2017)

Anahtar Kelimeler
Kleptografi,
Kriptografi,
RSA,
Kriptografik arka kapı

Özet: "Kriptografik bir sistemden, gizli bilgileri farkedilmeden ve sadece algoritmik değişiklerle çalabilme çalışmaları" olarak özetleyebileceğimiz *Kleptografi* alt disiplinini incelediğimiz bu çalışmada, RSA şifreleme sistemine karşı kurgulanmış kleptografik atak senaryolarını, ilgili algoritmaları ve bu algoritmaların, atak barındırmayan standart algoritmaların gerçeklenmesi ile oluşan sonuçların karşılaştırmalı analizleri ele alınacaktır. Özellikle bu çalışmalarda, atakların bazıları implemente edilmiş ancak standart algoritma ile oluşacak davranış farkını gösterebilecek yeterli analizler yapılmamıştır. Bu çalışmada atakların ayırt edilebilmesi için yeterli olacak istatistiksel testler yapılmış ve oluşan sonuçlar analiz edilmiştir.

Security and Complexity Analysis for Kleptographic Backdoors of the RSA Encryption System

Keywords
Kleptography,
Cryptography,
RSA,
Cryptographic backdoor

Abstract: In this study, we propose the kleptographic attack scenarios for the RSA encryption scheme, related algorithms, and their complexity comparison analysis with the standard RSA encryption by studying the problem of *Kleptography*, which we can summarize as "stealing confidential information from a cryptographic by only maliciously manipulating the cryptographic algorithms". In particular, the attacks were already implemented, however, they are mostly not compared with the standard algorithms. In this work, we performed sufficient statistical tests and analyzed the differences of the attacks with the standard algorithms.

1. Giriş

Kriptoloji uzun yıllardır insanoğlu tarafından mahrem bilginin saklanması amacıyla kullanılmıştır. Bilişim teknolojilerinin ve dolayısıyla internet teknolojilerinin yaygınlaşmasıyla kriptoloji bilimi daha fazla önem kazanmıştır. Şifreleme sistemlerinin geliştirilmesi ve bu sistemlere karşı yapılan saldırı mekanizmalarının araştırılması 1980'lerden sonra üstünde çokça uğraşılan konulardan olmuştur.

Aynı süreçte bilim adamları şu soruyu sormuşlar ve cevap aramışlardır: "Bir şifreleme sistemini üretirken veya gerçeklerken, kullanımı sırasında saldırgan tarafına mahrem bilgiyi sızdırabilecek şekilde sistemler tasarlanabilir mi?". Bu soru, etik olarak üzerinde çalışılmaması gereken bir konu gibi görünse de şifreleme sistemlerinin kritik yerlerde kullanılması hasebiyle kritik bilgiye sahip kurum ya da kuruluşların hatta hiçbir devletin uzak duramayacağı kadar hassas bir konudur. Dolayısıyla bu konunun incelenmesi ve varsa tedbirlerinin alınması gerekmektedir.

Saldırgan tarafına şifrelenen bilgiyi sızdıran bir sistemi, arka kapı barındıran bir sistem olarak görebiliriz. Ek olarak çalınacak bilgiyi sadece saldırganın çalabileceği şekilde sızdıracak ve aynı zamanda arka kapının farkedilmemesini de sağlayacak atak mekanizmaları çalışmaları, *Kleptografi* çalışmalarının bütününü oluşturmaktadır.

Bu çalışmada RSA Şifreleme Sistemine karşı kurgulanmış bazı kleptografik ataklar incelenmiş. Atakların yer aldığı çalışmalarda, güvenlik durumları yeterli olarak belirtilmiş ancak bu durum sadece teorik olarak ifade edilmiştir. Bu çalışmada bu atakların istatistiksel testler sonucunda güvenli olmadıkları gösterilmiştir.

1.1. Kleptografi Çalışmaları

"Mahrem bilgiyi, güvenli ve farkedilmeden çalabilme çalışmaları" olarak özetleyebileceğimiz Kleptografi tanımını ilk olarak Adam Young ve Moti Yung 1996 yılındaki [1] çalışmalarında yapmışlardır. Ancak yazarların kendi çalışmalarında da belirttikleri gibi bu fikrin temeli, Gus

*İlgili yazar: emreceran@sehir.edu.tr

Simmons'ın 1984 yılında [2]'de öne sürdüğü "subliminal kanallar" fikrine dayanmaktadır. Simmons bu çalışmada normalmiş gibi görünen bir iletişim hattında, aslında başka bir haberleşmenin gerçekleşebileceği fikrini öne sürmüştür. Aynı çalışmada bazı dijital imza algoritmalarında, rastgele seçilmesi gereken değerleri mesaj taşıyabilecek şekilde belirleyerek subliminal kanallar için uygulama örnekleri de verilmiştir.

Kleptografi'nin tanımını yapan ve bu alanda en çok çalışma yayınlayan isimler olarak karşımıza çıkan Adam Young ve Moti Yung; 1996 yılında yayınladıkları [1] çalışmalarında Kleptografi tanımıyla beraber RSA [3] ve ElGamal Şifreleme [4], Dijital İmza Algoritması (DSA) [5] ve Kerberos protokolüne [6] karşı kleptografik ataklar sunmuşlardır. Bu çalışmadan bir yıl sonra [7] çalışmalarında Kleptografik bir atak için güvenlik seviyeleri belirlemişler ve Diffie-Hellman anahtar değişim protokolü [8] için kleptografik atak sunmuşlar ve RSA için sundukları kleptografik atağı geliştirmişlerdir.

2003 yılında Crépeau ve Slakmon [9] çalışmalarında RSA şifreleme sistemine karşı arka kapılar kurgulamışlardır. Çalışmalarındaki arka kapılardan bir tanesi, Coppersmith'in [10] çalışmasında sunduğu, RSA şifrelemede gizli asalların hepsini bilmek yerine, sadece bitsel gösterimlerinin yarısını bilerek gizli anahtarın ele geçirilebileceğini ispatladığı Kısmi Bilgi Atağı'nın kullanıldığı, ilk arka kapı çalışması olarak ön plana çıkmaktadır. Bu atağın detaylı analizi ve çalışmadaki diğer ataklar hakkında özet bilgiler Bölüm 4.3'te ele alınacaktır.

Young ve Yung ikilisi 2004 yılında Kleptografik atak senaryoları ve ilgili diğer başlıkları detaylıca ele aldıkları "Malicious cryptography: Exposing cryptovirology" isimli kitaplarını yayınlamışlardır [11]. Bundan sonra yazarlar 2006, 2007 ve 2010 yıllarında eliptik eğrilerin kullanıldığı atak çalışmaları üzerine üç çalışma sunmuşlardır [12–14]. Bu süreçte kleptografi veya özel olarak arka kapılar kurgulama alanında çalışmalar sunan diğer yazarlar, daha çok kriptografik protokollere arka kapı çalışmaları ve arka kapı içeren bir sisteme karşı önlem alma çalışmaları yapmışlardır.

Kriptografik protokollere karşı arka kapı ataklarında SSL/TLS protokolü pratikteki yaygın kullanımından dolayı ön plana çıkmaktadır. Bu yönde Golebiewski ve arkadaşlarının [15] çalışmaları, Goh ve arkadaşlarının [16] çalışmaları ve Young ve Yung'ın [12–14] çalışmalarında SSL/TLS protokolüne karşı kleptografik ataklar bulunmaktadır.

Gogolewski ve arkadaşları 2006 yılında yayınladıkları [17] çalışmalarında *Elektronik Seçim* sistemlerine, 2008 yılındaki [18] çalışmalarında ise *Online Açık Artırma* sistemlerine karşı kleptografik ataklar sunmuşlardır.

2013 yılına kadar bu çalışmalar, kriptologların uğraştığı ve sadece teorik olan çalışmalar zannedilirken; *The New York Times* [19] ve *The Guardian* [20] gazetelerinde, Edward Snowden'in ortaya çıkarttığı gizli NSA belgelerine dayanarak yayınlanan haberlere göre NSA, şifreleme sistemleri üreticisi firmalarla ürettikleri sistemlerin, sadece NSA'in (Amerikan Ulusal Güvenlik Teşkilatı) faydalanabileceği şekilde zafiyet barındırmaları için yıllık 250 Milyon dolarlık bir projeyi uyguladığını açıklamışlardır. Bu belgede ayrıca NIST'in (Amerika Ulusal Standartlar ve Teknoloji Enstitüsü) 2006 yılında yayınladığı *"Special Publication 900-90"* [21] rastgele sayı üreteçleri önerilerine de atıf yapılmaktadır [22]. Bu belgede yer alan üreteçlerden özellikle bir tanesi dikkat çekmektedir. Bu üreteç *Dual_EC_PRBG* olarak isimlendirilen eliptik eğrilerin kullanıldığı rastgele sayı üretecidir ve araştırmacılar tarafından, Snowden'in ifşa ettiği belgeler yayınlanmadan önce de bu üretecin yavaş olduğu ve yavaş olmasına rağmen standartlaşmış olmasının zafiyet barındırmasından kaynaklandığı hakkında çalışmalar yapılmıştır [23–25]. Sonuç olarak, Snowden haberleri ile de birleştirildiğinde bu üretecin arka kapı barındırdığı söylenebilir.

NSA'in arka kapıları haberinden sonra araştırmacılar, kleptografi ile daha çok ilgilenmişler ve çoğunlukla arka kapı barındıran veya barındırabilecek bir sisteme karşı önlemler ile ilgilenmişlerdir. Bunlara örnek olarak Mironov ve arkadaşlarının 2015 yılında öne sürdükleri *Kriptografik Ters Güvenlik Duvarları (Cryptographic Reverse Firewalls)* sistemi sunmuşlardır [26]. Bu sisteme göre güvenlik duvarı normal bir güvenlik duvarı gibi sistemin dışarıdan gelen paketleri kontrol etmek yerine, içeriden çıkan paketleri rastgelelik katarak arka kapı yerleştirmiş olabilecek bir saldırgana karşı önlem almaktadır. Diğer bir çalışma ise yine 2015 yılında Russel ve arkadaşlarının *Cliptography* ismini vererek öne sürdükleri sistemdir [27]. Yazarlar bu çalışmada, rastgele sayı üreteçleri olarak tek yönlü fonksiyonları incelemişler ve herhangi bir sistemin arka kapı barındırdığı ön kabulü altında alınabilecek önlemler üzerine çalışmalarda bulunmuşlardır.

1.2. Atak Senaryosu

Şifreleme sistemlerinin kullanıcılarından, şifreledikleri mahrem bilgileri çalabilmeyi amaçlayan bir kapalı-kutu (incelemeye kapalı veya zorlaştırılmış sistemler) şifreleme sistemleri üreticisini saldırgan olarak düşünelim. Saldırgan, sistemlere arka kapılar kurgulamaya çalışmaktadır. Öncelikli hedefi arka kapıyı tespit edilemeyecek şekilde üretmektir. Bunun için de kullandığı sistemde arka kapı olduğundan şüphelenen bir kullanıcının ilk kontrol edeceği gösterge olan çıktıların, normal bir sistemin çıktıları ile uyuşmasını sağlamaya çalışacaktır. Yani arka kapı barındıran sistemin çıktılarının olasılık dağılımını, standart bir sisteminki ile örtüşmesini sağlayacak şekilde çalışacaktır. Ancak kullanıcının çıktıların dağılımından sonra bakabileceği bir diğer gösterge çalışma zamanı olduğundan, arka kapı barındıran sistemi, normal sistemin çalışma zamanı ile aşırı farklılıklar olmadan kurgulaması gerekecektir.

Yukarıdaki senaryo ile beraber Young ve Yung [1] çalışmalarında şöyle bir uygulama senaryosu da sunmuşlardır. Bu senaryoya göre arka kapı devletin (yasalara göre talep edilmesi halinde) istediği zaman kullanabileceği şekilde tasarlanacaktır. Buna göre gerektiği zaman dinlenebilecek böyle arka kapı barındıran sistemleri herkesin kullanması sağlanacak ve gerektiği zaman devlet, istediği sistemi dinleyebilecektir. Ancak bu durumda önemli bir güvenlik unsuru daha ortaya çıkmaktadır. Bu da böyle arka kapıların sadece ve sadece saldırgan tarafından kullanılabilmesi gerekliliğidir. Yani bir arka kapı

başka sistemlerde bir şekilde (tersine mühendislik gibi tekniklerle) ele geçirilse bile diğer kullanıcıların mahrem bilgilerini çalabilmek ancak saldırgan tarafından mümkün olabilmelidir. Bu faktörleri de göz önünde bulunduran arka kapı barındıran sisteme tasarlayıcıları hem sistemin normal kullanım halindeki gerekli kriptografik güvenlik gerekliliklerini sağlayabilmesini hem de arka kapının normal bir sistemden ayırt edilememesi ve kullanıcının (saldırgan hariç herkese karşı) mahrem bilgisini muhafaza edebilmesini amaçlayacaklardır.

Bu senaryolarda bahsedilen güvenlik gerekçelerini de göz önünde bulundurarak kleptografik bir atağın sağlaması gereken özellikler, tanım olarak aşağıdaki gibi olacaktır.

1.3. Kleptografi

Kleptografi bu alanda en çok çalışma yayınlayan isimler olan Young ve Yung tarafından [1] eserlerinde, *"Bilgiyi subliminal ve asimetrik olarak çalabilme çalışmaları"* olarak tanımlanmışlardır. Burada iki önemli odak noktası karşımıza çıkmaktadır.

1. Kleptografik bir atak asimetrik olmalı: Çalınmak istenen bilgi saldırgandan başkası tarafından ele geçirilememelidir.

2. Kleptografik bir atak subliminal olmalı: Kurban yani kullanıcı, şifreleme sisteminde atak olup olmadığını farkedememelidir.

1.4. Kleptografik Bir Atağın Güvenliği

Kleptografik bir atağın güvenlik durumunu incelerken, normal bir kripto sistemin güvenliğini incelemekten biraz daha faklı düşünmemiz gerekecektir. Çünkü kleptografik bir atakta, biri diğerinin içine gizlenmiş iki ayrı şifrelemenin güvenliği ve bu sistemlerin birbiriyle uyumlu olması amaçlanmaktadır. Normal bir kripto sistemde en önemli hedef gizlilik; yani mahrem bilginin korunması iken kleptografik bir sistemde, gizliliğin yanında atağın deşifre olmaması için normal sistemle aynı ölçülebilir özelliklere sahip olmasına dikkat edilecektir. Kleptografik bir atak barındıran bir sistemin çıktılarının normal bir sistemle aynı özelliklere (çıktıların olasılık dağılımı) sahip olması (ayırt edilemezlik) incelenecek en önemli özelliklerdendir. Bir diğer ölçülebilir özellik ise sistemin çalışma zamanıdır. Çıktıların olasılık dağılımını normal sisteminkine benzetebildiğimiz taktirde sistemin atak barındırıp barındırmadığını test eden bir kullanıcı ikinci olarak bakabileceği tek özellik çalışma zamanının beklenen süre içinde gerçekleşip gerçekleşmeyeceğidir.

1.4.1. Atakların Simülasyonu ve Analizleri

İlerleyen bölümlerde incelenecek ataklarda, ilgili algoritmaları inceledikten sonra; bu algoritmaların gerekli kodlarını yazarak simülasyonlarını ve bu simülasyonlarla üretilecek çıktıların standart algoritmaların çıktıları ile kıyaslamalı analizlerine yer vereceğiz. Bu analizler çıktıların istatistiksel dağılımı ve anahtar üretim çalışma zamanı olarak iki madde halinde incelenecektir.

Üretilen Anahtarların İstatistiksel Dağılımı. İnceleyeceğimiz ataklarda üretilecek olan açık anahtarlar, gizli anahtarın tamamını veya gizli anahtarı elde edebilmeye yetecek kadarını, saldırganın ele geçirebileceği şekilde içinde barındıracaktır. Bu ise açık anahtarların standart algoritmalardan farklı dağılıma sahip olmasına sebebiyet vermektedir. Crepeau ve Salkmon [9] çalışmalarında, bu durumun [1] çalışmasındaki RSA atağında karşılaşıldığıunı aşağıdaki gibi eleştirmişlerdir; " PAP algoritması (Young ve Yung'ın [1] çalışmasındaki RSA atağına verdikleri isim) ile üretilen n açık anahtarlarının üst bitleri, rastgele seçilmiş asallarla sağlanmayacak şekilde düzgün rastgele dağılıma sahiptir. Örneğin; 512 bit rastgele seçilmiş iki asal sayı ile üretilen $n = pq$ açık anahtarı, %38 olasılıkla 1023 bit; %48 olasılıkla "10" ile başlayan 1024 bit ve %14 olasılıkla "11" ile başlayan 1024 bit uzunlukta olacaktır. Ancak bu durum [1] çalışmasındaki RSA atağında sağlanmamaktadır."
Asal sayılar teoreminin bir sonucu olan yukarıdaki dağılım, yönlendirilmemiş bir rastgele bit üreteci ile üretilen asalların çarpımı sonucu oluşacak olan tamsayıların dağılımı ile örtüşecektir. Bu çalışmada üretilen anahtarların dağılımını test etmek için yukarıdaki yöntemden faydalanılmıştır. Öncelikle rastgele seçilerek standart algoritmayla üretilen anahtarların dağılımları ölçülmüş, bu dağılımlar atak algoritmaları ile üretilen anahtar değerleri ile karşılaştırılarak, atakların ayırt edilemezliği test edilmiştir. Burada kullanılan rastgele bit üreteci kütüphanesinin yönlendirilmiş olabileceği sorusu akla gelebilir. Ancak ilerleyen bölümlerde de görülebileceği gibi atak algoritmalarında da saklanacak değerler, kısmen rastgele seçilmektedir. Dolayısıyla yönlendirilmiş bir rastgele bit üreteci olsa bile her iki durumu da etkileyecektir. Burada dikkat edilmesi gereken nokta karşılaştırılan değerleri üretmek için kullanılacak değerlerin ilk seçimi aynı kütüphaneyle yapılmaktadır.

Anahtar Üretim Çalışma Zamanı. Kleptografik bir atağın tespit edilebilmesini sağlayacak bir diğer parametrenin çalışma zamanı olduğunu daha önceden belirtmiştik. Simüle edilen ataklarda standart algoritma ile atak barındıran algoritmaların çalışma zamanları ölçülmüş ve aynı dağılım değerlerinde olduğu gibi karşılaştırmalı analizleri yapılmıştır. Burada şu hususu belirtmek de fayda olacaktır; çalışma zamanı, istatistiksel dağılım gibi sadece algoritmaya göre değişen bir parametre değildir. Belirleyen en önemli etmen algoritma olsa da diğer yandan simüle edilen bilgisayarın o anki durumu, işletim sisteminin o anki işlemleri gibi etmenler bu parametreyi etkileyecektir. Her ne kadar algoritmaların çalışma süreleri boyunca internet bağlılığı gibi faktörler aynı tutulmaya çalışılmış olsa da, yine de etkileyebilecek başka faktörler de olabilecektir.

Simulasyonların yapıldığı bilgisayarın detayları. Bu çalışmada verilen simülasyonlar, Intel Core2 Duo CPU P8800 işlemci ve 16GB RAM'e sahip, Windows 7 işletim sisteminde çalışan bir bilgisayarda, Python 2.7 program-

lama dili ile gerçekleştirilmiştir.

2. Gerekli Altyapı

Bu bölümde okuyucunun ihtiyaç duyabileceği gerekli matematiksel altyapı özet halinde ele alınacaktır. Bunun için öncelikle özet cebir bilgileri, sonrasında sayılar teorisi ile ilgili bazı tanım ve teoremler ele alınacaktır.

2.1. Cebir ve Sayılar Teorisi

Grup, *Halka* ve *Cisim* gibi cebirsel yapılar ile ilgili temel tanımlamaları ve kriptografide kullanılan ilgili temel teoremler, Asal Sayılar, ve Modüler Aritmetik konularında kriptografi ile ilgili temel tanım ve teoremler bu çalışmada ele alınmayacak ancak okuyucunun hatırlamakta zorlanabileceği bazı özel tanım ve teoremler bu bölümde ele alınacaktır. Yukarıda bahsedilen temel tanım ve teoremler ile ilgili detaylı bilgiler [28], [29] kaynaklarında bulunmaktadır. İnternet'te Türkçe olarak verilen kaynaklar ise Prof. Dr. Erhan Güzel Cebir Sayfası [30] ve Marmara Üniversitesi Fen-Edebiyat Fakültesi Cebir Ders Notları [31] sayfalarında bulunabilir.

Tanım 2.1 (Döngüsel grup, Mertebe)**.** Bir G grubunda $\forall b \in G$ için $b = \alpha^i$, $i \in \mathbb{Z}$ olacak şekilde bir $\alpha \in G$ elemanı bulunabilirse bu gruba döngüsel bir grup denir ve α elemanına bu döngüsel grubun üreteci denir.
G bir grup ve $a \in G$ olsun. Bir a elemanı için $a^t = 1$ sağlayan en küçük t değeri varsa bu t değerine a elemanının mertebesi denir ve $ord(a) = t$ ile gösterilir. Eğer böyle bir t değeri bulunmazsa a elemanının mertebesi ∞'dur denir.

Tanım 2.2 (Denklik Sınıfları, Çarpımsal Grup)**.** $n \in \mathbb{Z}^+$ pozitif tam sayı, $a,b \in \mathbb{Z}$ tam sayılar olmak üzere, n sayısı, $(a-b)$ farkını bölüyorsa a, b'ye mod n'de kongrüdür denir ve $a \equiv b \bmod n$ şeklinde gösterilir. Bir a tam sayısının mod n'de kongrü olan bütün tam sayıların oluşturduğu kümeye; a'nın mod n'de denklik (kalan) sınıfı denir ve $\overline{a} = \{x | x \in \mathbb{Z}, a \equiv x \bmod n\}$ şeklinde temsil edilir. mod n denklik sınıflarından oluşan küme $\mathbb{Z}_n$ ile gösterilir ve n'den küçük $\{0,1,\cdots,n-1\}$ tam sayılarının birbirinden farklı denklik sınıflarının oluşturduğu kümeye denir. $\mathbb{Z}_n^* = \{\overline{a} \in \mathbb{Z}_n |\ \text{EBOB}(a,n) = 1\}$ şeklinde tanımlanan kümeye, $\mathbb{Z}_n$'in çarpımsal grubu denir ve n'den küçük ve n ile aralarında asal olan tam sayıların denklik sınıflarının oluşturduğu grubu temsil eder. Özel olarak grubu belirleyen n sayısı asal ise $\mathbb{Z}_n^* = \{\overline{a} | 1 \le a \le n-1\}$ olacaktır [28].

Tanım 2.3 (Euler Phi Fonksiyonu)**.** $n \in \mathbb{Z}^+$ tam sayısı için, n'den küçük ve n ile aralarında asal olan sayıların sayısı $\varphi(n)$ ile gösterilir ve Euler Phi fonksiyonu adı verilir.

Euler Phi fonksiyonu özellikleri [28]:

1. p asal ise $\varphi(p) = p-1$
2. $EBOB(m,n) = 1$ ve $m,n \in \mathbb{N}$ ise $\varphi(mn) = \varphi(m)\varphi(n)$
3. $n = p_1^{e_1} p_2^{e_2} \cdots p_n^{e_n}$ ise

$$\varphi(n) = n\,(1-\frac{1}{p_1})\,(1-\frac{1}{p_2})\cdots(1-\frac{1}{p_n})$$

Bu çalışmada RSA şifreleme sistemine karşı kurgulanan kleptografik ataklar ele alınacaktır. RSA şifrelemede bir m mesajını şifrelemek için, p ve q rastgele seçilecek asal sayılar, $n = p \cdot q$ açık anahtar olmak üzere, diğer açık anahtar e ve gizli anahtar d değeri, Euler Phi $\varphi(n)$ bir fonksiyon olmak üzere;

$$d \equiv e^{-1} \bmod \varphi(n)$$

şeklinde belirlenmektedir.
RSA şifreleme sisteminde, açık anahtarlar (n,e) ve gizli anahtar d değerleri belirlendikten sonra; m mesajını şifrelemek için

$$c = m^e \bmod n$$

operasyonu kullanılmaktadır. Şifrelenmiş mesajı açmak isteyen kullanıcı

$$m \equiv c^d \bmod n$$

işlemiyle açık metine ulaşabilecektir. Daha açık ifadeyle açık anahtar e değeri, $EBOB(e,\varphi(n)) = 1$ olarak seçildiğinden $e \in \mathbb{Z}_n^*$ olur ve gizli anahtar modülo $\mathbb{Z}_{\varphi(n)}$ üzerinde $d = e^{-1}$ olarak seçilir. Yani öyle bir $k \in \mathbb{Z}$ vardır ki $ed = k\varphi(n) + 1$ olur.
Bu işlemin doğru olmasının sebebi aşağıda vereceğimiz Euler Teoremi'dir.

Teorem 2.4. *$n \ge 2$ tam sayı olmak üzere;*

1. *(Fermat'ın Küçük Teoremi [28]): Her p asal sayısı için eğer $EBOB(a,p) = 1$ ise $a^{p-1} \equiv 1 \bmod p$.*
2. *(Euler Teoremi [28]): $a \in \mathbb{Z}_n^*$ ve $EBOB(a,n) = 1$ ise $a^{\varphi(n)} \equiv 1 \bmod n$.*

Euler Teoremi'ni kullanarak RSA şifreleme sisteminde şifrelenmiş mesajın çözümünün açık metine eşit olacağını aşağıdaki gibi görebiliriz;

$$\begin{aligned} Dec_d(Enc_e(m)) &= Dec_d(m^e \bmod n) \\ &= (m^e)^d \bmod n = m^{ed} \bmod n \\ &= m^{1+k\varphi(n)} = m. \end{aligned}$$

Yukarıdaki denklemde $Enc_e(m)$ notasyonu, m mesajını e açık anahtarıyla şifreleme işlemini; $Dec_d(c)$ ise c şifreli metninin d gizli anahtarı ile şifre çözme işlemini temsil etmektedir.

3. RSA Şifreleme Sistemi

Kleptografi alanında, akademik çalışmalara baktığımızda, en fazla atağın kurgulandığı sistem olarak RSA şifreleme sistemini görebiliriz. Bunun sebebi RSA şifreleme sisteminin yaygın olarak kullanılmasıdır. Güvenliği *Çarpanlara Ayırma Problemi*nin büyük sayılardaki hesaplama zorluğuna dayanan RSA Şifreleme sistemine kurgulanmış kleptografik atakları ele alacağımız bu çalışmada, ilgili atakların algoritmalarının incelenmesinden sonra simülasyonlarının analizi de işlenecektir. Ataklara geçmeden önce RSA şifreleme sistemini kısaca görmekte fayda olacaktır. Birbiriyle güvenli haberleşmek isteyen iki taraftan biri açık ve gizli anahtarlar üretecek ve mesaj göndermek isteyen

taraf diğer kullanıcının açık anahtarıyla göndermek istediği mesajı şifreleyerek gönderecektir. Şifreli mesajı alan kullanıcı gizli anahtarıyla şifreli metni çözerek açık mesaj değerine ulaşabilecektir. Gizli ve açık anahtar üretmek isteyen kullanıcı, aşağıdaki algoritmayı kullanacaktır.

RSA Anahtar Üretimi (k): [3]
Girdi:
k güvenlik parametresi (üretilecek asalların bit uzunluğu)
Çıktı:
Açık anahtar: (n,e) öyle ki $n \in \{\{0,1\}^{2k-1}, \{0,1\}^{2k}\}$; $1 < e < \varphi(n)$ ve $EBOB(e,\varphi(n)) = 1$
Gizli anahtar: d öyle ki $d \equiv e^{-1} \bmod \varphi(n)$
($\varphi(n)$: Euler Phi fonksiyonu)

1. $p,q \in_R \{0,1\}^k$ asal sayıları seç. $(p \neq q)$
2. $\varphi(n) = (p-1)(q-1)$ hesapla.
3. e seç öyle ki $1 < e < \varphi(n)$ ve $EBOB(e,\varphi(n)) = 1$
(e açık anahtarının sabit bir sayı olması istendiğinde bu adım geçilir.)
4. $d = e^{-1} \bmod \varphi(n)$ hesapla
(e açık anahtarının sabitlenmesi istendiğinde q asalı bu adımda uygun d bulana kadar rassal bir algoritma ile üretilir.)
5. (n,e) açık anahtar, d gizli anahtar çıkart.

Şekil 1. RSA Şifreleme Anahtar Üretim Algoritması

Yukarıdaki algoritmayla üretilen anahtarlardan (n,e) açık anahtarları yayımlanır ve gizli olan d anahtarı saklanır. Açık anahtarlarla bir m mesajını $(m < n)$ şifrelemek isteyen bir kullanıcı

$$c \equiv m^e \bmod n$$

hesaplayarak c şifreli metnini elde eder ve anahtar sahibine gönderir. Mesajın ulaşması gereken kullanıcı, d gizli anahtarıyla

$$m = c^d \bmod n$$

hesaplayarak açık metine ulaşabilir.

3.1. Algoritmanın Simülasyonu ve Analizi

Daha önce belirtiğimiz gibi kapalı kutu bir şifreleme sisteminde, kleptografik bir atak olup olmadığını test etmek için bakılabilecek ilk kriter sistemin ürettiği çıktıların, standart algoritmaların ürettiği çıktıların olasılık dağılımlarının uyumlu olup olmadığıdır. Diğer kriter ise tersine mühendislikle sistemde çalışan kodun görülmesi ancak bu işlem çok daha zahmetli ve bazı tekniklerle zorlaştırılabilmektedir. Bu çalışmada incelenecek ataklarda, atak algoritmaları verildikten sonra ataklar simüle edilecek ve ürettiği çıktılar standart algoritmaların üreteceği çıktılar ile karşılaştırılacaktır. RSA şifreleme için atakları incelerken karşılaştırma kriteri olarak kullanmak üzere standart algoritma ile, ataklarda yapacağımız gibi 150 adet anahtar üretip bu anahtarların olasılık dağılımlarını ve anahtar üretimi için gerekli çalışma zamanlarını bu bölümde inceleyeceğiz.

Algoritmaların gerçeklendiği bilgisayarın fiziksel durumu, işletim sistemi ve gerçeklendiği platform gibi detaylar Bölüm 1.4.1'da yer almaktadır.

3.1.1. Çalışma Zamanı

İlerleyen bölümlerde ele alınacak atakların, simülasyonlar sonucu oluşacak çıktı değerleri ile karşılaştırmak için üretilen 150 adet standart, arka kapı barındırmayan anahtarın çalışma zamanı ortalamaları Tablo 1'de verilmiştir. Tablodaki değerler, 3 ayrı anahtar boyu; 256, 512, 1024 için (sırasıyla asal uzunlukları 128, 256, 512 ile temsil edilecek ve anahtar boyları güvenlik için yetersiz olsa da analiz için yeterli olacaktır) seçilen örneklerin çalışma zamanlarını göstermektedir.

Tablo 1. Zaman Tablosu:

k:asal bit uzunluğu	RSA
128	0,07
256	1,01
512	27,97

150 adet anahtarın üretim zamanı ortalamaları (sn.)

3.1.2. Açık anahtar dağılımı

Bu bölümde üretilen anahtarları analiz edebilmek için muhtemel tüm anahtarlar kümesini büyüklüklerine göre 3 parçaya böldük ve üretilen anahtarların bu 3 ayrı kümede olma istatistiklerini belirledik. Bu işlemi bitsel gösterimindeki şu analizle yapacağız.

Analiz için üretilen anahtarlarda, k bit uzunluğunda p,q asalları ile $n = pq$ olarak üretilen açık anahtar mod değerleri, ya $2k-1$ bit uzunluğunda ya da $2k$ bit uzunluğunda olacaktır. Analiz için sadece bit uzunluğunda göre ayırmak yerine ikinci bölgeyi yani $2k$ bit uzunluğunda olanların kümesini de iki parçaya böldük ve bitsel gösterimi "11" ile başlayan $2k$-bit uzunluğundakiler ve "10" ile başlayan $2k$ bit uzunluğundakiler olarak tasnif ettik. Bu sayede çıkarılacak anahtarları büyüklüklerine göre 3 grupta kategorize edebileceğiz.

Üretilen 150 adet dürüst RSA anahtarı için, n değerinin olasılık dağılım yüzdeleri Tablo 2'de verilmiştir.

Tablo 2. n ahatarı dağılımı tablosu

	RSA		
Bit uzunluğu k	$2k-1$	$2k$ "10" ile başlayan	$2k$ "11" ile başlayan
128	%38	%56	%6
256	%36	%51	%12
512	%37	%48	%14

150 adet n açık anahtarın dağılım yüzdeleri

4. RSA İçin Kleptokrafik Ataklar

4.1. YY96: Young ve Yung'un RSA için İlk Kleptografik Atağı

Bu bölümde, Young ve Yung ikilisinin 1996 yılında [1] çalışmalarında sundukları, RSA için ilk kleptografik atağı ele alacağız. Bu atakta saldırgan, kullanıcının şifreleme sistemine kendi (N,E) RSA açık anahtarlarını yerleştirmiş ve bunlarla kullanıcının üretilecek gizli asalı p'yi şifreleyerek, n açık anahtar değeri içerisinde yayınlanmasını sağlayacaktır.
Atakta kullanılacak sabitler ve fonksiyonların detayları aşağıdaki gibidir:

- **Saldırganın anahtarları:** Atak algoritmasında (N,E) saldırganın RSA açık anahtarını ve D değeri gizli anahtarını temsil etmektedir. Açık anahtarlar, atak algoritması içinde yer alacak; ancak gizli anahtar sadece saldırganda bulunacaktır.

- **Rastgeleleştirme Fonksiyonları:** Algoritmada F ve G ile temsil edilecek iki fonksiyon kullanılacaktır. Bu fonksiyonlar, manipüle edilecek olan değerlerin rastgeleliğini ve çıktıların istenilen aralığa düşmesini garantileyerek, gizli p asalını çalabilmeye uygun hale getirmeye yarayacak fonksiyonlardır. Bunlar simetrik şifreleme algoritmaları olabilir ancak tersi alınabilir olmaları gerektiğinden (anahtar geri kazanırken tersleri kullanılacaktır) kriptografik özet fonksiyonlar olamayacaklardır.

- B_i **sınırları:** Algoritmada kullanılacak olan B_1 değeri üretilecek p asalının saldırganın N modundan küçük olmasını sağlamakta iken; B_2 değeri ise q asalının, asal olmasını sağlamaya çalışırken kullanılacak döngünün kurulabilmesinde işimize yarayacaktır. Young ve Yung bu çalışmalarında atağın implementasyonunu da yapmışlar ve B_1 ve B_2 değerlerini sırasıyla 16 ve 512 olarak seçmişlerdir.

- F **ve** G **rastgeleleştirme fonksiyonu anahtarları:** i ve j değerleri, saldırgan tarafından seçilen sabit bir K değeri ile birlikte, rastgeleleştirme fonksiyonlarının anahtarlarını belirlemektedirler. Bu değerler B_i değerleri ile sınırlandırıldığından saldırgan gizli anahtarı elde etmek için i ve j değerlerini kullanmak istediğinde hangi değerlerin atak esnasında kullanıldığını bilmese de tahmin edebilecek yani muhtemel bütün i ve j değerlerini deneyerek istediğini elde edebilecektir. Algoritmada $||$ notasyonu ile bitsel dizileri uçuca ekleyerek birleştirme işini yapacak olan birleştirme (concatenation) operatörü temsil edilmektedir.

Bu algoritmada; Adım 3'te sızdırılmak istenen p asalı saldırganın açık anahtarıyla şifrelenmektedir. Adım 4'deki döngüde ise p asalının rastgeleleştirilmiş ve şifrelenmiş hali rastgele bir bit dizisine ekleme operasyonuyla, kullanıcının açık anahtarı n'e dönüştürülmektedir. Bu X değeri yine Adım 4'de p asalına bölünerek q değeri elde edilir.

KleptoAnahtarÜreteç(k): [1]
Girdi:
k: Gizli asalların bit uzunluğu.
Çıktı:
Açık anahtar: (n,e) öyle ki $n \in \{\{0,1\}^{2k-1};$ $\{0,1\}^{2k}\}$, $1 < e < \varphi(n)$ ve $EBOB(e,\varphi(n)) = 1$
Gizli anahtar: d öyle ki $d \equiv e^{-1} \bmod \varphi(n)$
Rastgeleleştirme fonksiyonları: F,G
rastgeleleştirme için kullanılacak simetrik şifreleme fonksiyonları (ör: DES, XTEA, AES).
Gömülü Değerler: Saldırgan RSA anahtarları: $N \in \{0,1\}^k$ ve $E < \varphi(N)$ açık anahtarları.
K anahtarı: F ve G fonksiyonlarında kullanılacak anahtar değeri.

1. $p \in_R \{0,1\}^k$ asalı seç
2. $i = 0$'dan B_1'e kadar;
   $p' \leftarrow F_{K+i}(p)$ hesapla
   $p' < N$ ise bırak değilse i'yi 1 arttır.
3. $p'' := (p')^E \bmod N$
4. $j = 0$'dan B_2'e kadar;
   $p''' \leftarrow G_{K+j}(p'')$ hesapla
   $rand \leftarrow k - bit$ uzunluğunda rastgele bit dizisi
   $X \leftarrow (p''' \,||\, rand)$
   $q := X/p$
   q asal ise sonraki adıma geç, değilse j değerini 1 arttır
   Adım 1'e dön
5. $n \leftarrow p.q$; $\varphi(n) \leftarrow (p-1)(q-1)$; $e = 17$
6. $(e,\varphi(n)) = 1$ ise $d = e^{-1} \bmod \varphi(n)$ hesapla değilse e'yi 2 artır
7. (n,e,d) çıkart

Şekil 2. Kleptografik RSA Anahtar Üretim Algoritması [1]

Adım 6'da kullanıcının açık kuvveti olan e değeri başlangıçta 17 olarak belirlenir ve daha sonra açık anahtar modülü n değerinin Euler $\varphi(n)$ fonksiyonuyla aralarında asal oluncaya kadar 2 arttırılır.

Kullanıcının açık anahtar olarak yayınladığı n değerini ele geçiren ve D gizli anahtar değerine sahip saldırgan aşağıdaki *Anahtar Ele Geçirme* algoritmasını kullanarak $n = pq$ asallarına ayırabilecek ve buradan gizli anahtar olan d değerine ulaşabilecektir.

AnahtarEleGecir algoritmasında, kullanıcının açık olarak yayınlanan n modül değerinden gizli asal değerlere ulaşmaya çalışılacaktır. Bunun için öncelikle n değerinin bitsel gösteriminin en düşük (sağdan) k tane biti atılacak ve kalan bitlerin tam sayı değeri U değişkenine atanacaktır. Daha sonra bu U değeri G fonksiyonunun tersi ile muhtemel bütün $K+j$ anahtarları kullanılarak p'' değerleri elde edilir. Bulunan bu değerler D kuvvetiyle şifre çözme

AnahtarEleGeçir(n): [1]
Girdi: Açık anahtar: n öyle ki $n \in \{\{0,1\}^{2k-1}, \{0,1\}^{2k}\}$.
Çıktı: $p \in \{0,1\}^k$ asal sayısı.
Operatörler: $|n|$: n sayısının bitsel uzunluğu
$n\rceil^t$: n sayısının en üst (sol) t biti

1. $U := n\rceil^{|n|-k}$.
2. $L_1 := \{p'' \leftarrow G_{K+j}^{-1}(U) : K = \text{ sabit }, j = 0, \cdots, B_2 - 1\}$
3. $L_2 := \{p' \leftarrow (p'')^D \bmod N : p'' \in L_1\}$
4. $L := \{p \leftarrow F_{K+i}^{-1}(p') : K = \text{ sabit }, i = 0, \cdots, B_1 - 1\}$
5. $c \in L$ için;
   $c|n$ ise $p = c$ çıkart
   böyle eleman bulunamazsa diğer adıma geç
6. $U = U + 1$ yap ve Adım 2'ye dön

Şekil 3. Gizli Anahtar Ele Geçirme Algoritması [1]

yapıldıktan sonra F fonksiyonunun tersi ile muhtemel bütün $K+i$ anahtarları kullanılarak, muhtemel bütün p değerlerine ulaşılır. Bulunan bu değerler arasında n modülünü bölebilen bir değer varsa, gizli p asalı bulunmuş olur. Eğer böyle bir p değeri bulunamazsa *KleptoAnahtarÜreteç* algoritmasında X'in p asalına bölünmesi esnasında ödünç bit alınmış olabileceğinden $U \leftarrow U + 1$ olarak atanır ve 2. Adım'a dönülür.
Algoritma düzgün çalışacaktır çünkü yayınlanan n açık anahtarı;

$$X = G(F(p)^E \bmod N)||rand$$

ve q asalı $q = X/p$ olmak üzere;

$$n = pq$$

$$n = \frac{G(F(p)^E \bmod N)||rand}{p} \cdot p$$

$$n = G(F(p)^E \bmod N)||rand$$

olarak belirlenecektir. $U = n\rceil^{|n|-k}$ olmak üzere;

$$G^{-1}(U) = G^{-1}(G(F(p)^E \bmod N)) = F(p)^E \bmod N$$

$$(G^{-1}(U))^D \bmod N = (F(p)^E)^D \bmod N = F(p)$$

$$F^{-1}((G^{-1}(U))^D \bmod N) = F^{-1}(F(p)) = p$$

Sonuç olarak;

$$p = F^{-1}((G^{-1}(n\rceil^{|n|-k}))^D \bmod N)$$

olacak şekilde anahtar üretim algoritmasıyla üretilen n değerinden, anahtar ele geçirme algoritmasıyla p değerine ulaşılabilecektir.

4.1.1. Atağın Simülasyonu ve Analizi

Bu bölümde atak algoritmalarının gerçeklenmesi ve üretilen çıktıların standart algoritmalar ile karşılaştırmalı analizleri yapılacaktır. Bunun için; klepto anahtar üreteç ve anahtar ele geçirme algoritmalarını gerçeklenmiş ve bir önceki bölümde standart (atak barındırmayan) RSA anahtar üretiminde olduğu gibi 150 adet anahtar üretip bu anahtarların gizli değerlerini ele geçirerek atak simüle edilmiştir. Bunun sonucunda üretilen anahtarlar için rastgele bir mesajı şifreleme/şifre çözme başarıları ve açık anahtardan, gizli anahtarı ele geçirebilme başarı oranları beklendildiği üzere %100 olarak gözlemlenmiştir. Algoritmaların gerçeklendiği bilgisayarın fiziksel durumu, işletim sistemi ve gerçeklendiği platform gibi detaylar Bölüm 1.4.1'de yer almaktadır.

- **Çalışma Zamanı:** Simüle edilen atak için üretilen 150 adet anahtar değerinin ve bir önceki bölümde analiz edilen standart RSA anahtar üretiminin karşılaştırmalı anahtar üretim zamanları, Tablo 3'de verilmiştir. Bu tabloda 512-bit anahtarlar için gereken ortalama zamanın dürüst anahtar üretiminden fazla çıkması ile anahtarların şifreleme/şifre çözme başarıları arasında ters bir orantı oluşturulabilir. Yani anahtar üretim algoritmasında B_i sınırları azaltılarak, çalışma zamanı daha iyileştirilebilir ancak bu işlem, üretilen anahtarlardan ele geçirebilme başarısının da düşmesine sebep olabilir.

Tablo 3. Çalışma Zamanı

asal.bit.uzunluğu:k	RSA	[1]
128	0,07	0,11
256	1,01	1,17
512	27,97	42,15

150 adet anahtarın üretim zamanı ortalamaları (sn.)

- **Açık Anahtarın Dağılımı:** RSA anahtar üretiminde k-bit asallar seçerek üretilen açık anahtarın n modül değeri, ya $2k-1$ bit, ya da $2k$ bit uzunluğunda olacaktır. '$2k$-bit uzunluktaki değerleri, tekrar gruplandırmak istersek, bitsel gösterimi "10" ile başlayan $2k$ bit uzunluktaki ve "11" ile başlayan $2k$ bit uzunluktakiler olarak ayırabiliriz. Böylece üretilen anahtarları büyüklüklerine göre 3 ana grupta sınıflandırmış oluyoruz. Bu durum analiz için yeterli olacaktır. Ancak daha kesin analizler için bir adım daha ileri götürülerek 3. bitlerine göre de bir analiz yapılabilir.

  Atak barındıran anahtar üretim algoritması ve dürüst RSA anahtar üretim algoritması ile üretilen anahtarların, yukarıda bahsettiğimiz kriterlere göre sınıflandırması Tablo 4'de bulunabilir. Tablodan da görülebileceği gibi KleptoAnahtarÜreteç algoritmasının çıktıları, dürüst anahtar üretimi algoritmasına göre ayırt edilemezliği sağlamamaktadır. Sonuç olarak atak barındıran algoritmaya sahip sistemin çıktıları, teste tabi tutulduğunda sistemin atak barındırdığı ortaya çıkacaktır.

Tablo 4. Açık Anahtar Dağılım Tablosu

	RSA			[1]		
Bit Uzunluğu k	$2k-1$	$2k$ "10"	$2k$ "11"	$2k-1$	$2k$ "10"	2k "11"
128	%38	%56	%6	%31	%30	%38
256	%36	%51	%12	%32	%32	%34
512	%37	%48	%14	%34	%33	%32

150 adet n açık anahtarın dağılım yüzdeleri

4.2. YY97: Young ve Yung'un RSA için Daha Güçlü Kleptografik Atak

Bölüm 4.1'de ele aldığımız atağın yayınlanmasından bir sene sonra Young ve Yung [7] eserlerinde bu atağın neden geçersiz olduğunu ve daha güçlü hale getirmek için atakta nasıl değişiklikler yapılması gerektiğini açıklamışlardır.
Önceki bölümde sunduğumuz atağı ele alacak olursak. Kullanıcı bir şekilde şifreleme sisteminde arka kapı olduğundan şüpheleniyor olsun ve kontrol etmek istesin. Başka cihazlardan tersine mühendislik gibi tekniklerle elde edilen atak algoritması ve saldırganın açık anahtarları ile kullanıcı kendi sistemini test edebilecektir. Bunun için öncelikle (n,e) açık ve d gizli anahtarlarından $n = pq$ çarpanlarına ulaşacaktır. Verilen açık ve gizli anahtarlardan, p ve q çarpanlarına nasıl ulaşılabileceği Dan Boneh'nin [32] çalışmasında bulunabilir.
p ve q asal çarpanlarına ulaşan bir kullanıcı, p asalını aynen kleptografik sistemin üreteceği gibi kullanarak n ve q değerlerini üretecektir. Ürettiği değeri kendi n modunun en üst $|n|-k$ bitiyle karşılaştırarak, sistemde arka kapı olup olmadığını tespit edebilecektir.
Young ve Yung aşağıdaki bölümde verilecek atak mekanizmasında bu sorunu gidermişlerdir, bununla beraber seçilen asalın saldırganın N açık mod değerinden küçük olması gerekliliği sonucu, asalların belli bir bölgeye sıkışabilecek olması sorununu gidermek adına ikili çalışmalarında PBRM (Olasılıksal Eğilim Kaldırma Yöntemi) adını verdikleri bir metod önermişlerdir. Ataktan önce bu metodu görelim.

4.2.1. Olasılıksal Eğilim Kaldırma Yöntemi (Probabilistic Bias Removal Method PBRM)

Bir önceki bölümde ele aldığımız atak için aşağıda anlatacağımız gibi, değerlerde bir yönlenme sorunu oluşmaktadır. Young ve Yung [7] çalışmalarında bu sorunu şu şekilde açıklamışlardır. İlk adımda seçtiğimiz p asalı, saldırganın N açık anahtarından küçük olması gerekmekteydi. Ancak bu durumda saldırgan ya N değerini olabildiğince büyük seçecek, ya da seçilen p asalları belli bir bölgede (istenilen aralıkta küçük değerler alacak şekilde) birikecektir. Dolayısıyla sonuçta oluşan n açık anahtar değeri de olması gereken dağılımı sergileyemeyecektir.
Herhangi bir aralıkta verilmiş rastgele bir değer için, bu değeri girdi olarak alıp, daha geniş bir kümeye aynı dağılımı koruyarak transfer etmek istediğimizi düşünelim. Örneğin; $[1,R]$ aralığında düzgün dağılıma sahip verilmiş x değeri için, x' değerini $2R > S$ olmak üzere $[1,S]$ aralığında düzgün dağılıma sahip olacak şekilde elde etmek istiyoruz. Bu işlem, Şekil 4'de tanımlanan PBRM fonksiyonu kullanılarak gerçekleştirilebilir.

Klepto anahtar üretim algoritması p değerini rastgeleleştirdikten sonra PBRM fonksiyonundan geçireceği için, gizli anahtarı ele geçirmek için PBRM fonksiyonunun tersini kullanmamız gerekecektir. Bu fonksiyon aşağıdaki gibi basitçe hesaplayabiliriz.

$$\mathsf{PBRM}^{-1}(R,S,x') = \begin{cases} x = x' & x' < R \\ x_{1,2} = x', S - x' & x' \geq R \end{cases}$$

PBRM(R,S,x): [7]
Girdi: S, $R \in \mathbb{Z}$, $R \in (S/2,\ S)$, $x \in \{0,\ \ldots,\ R-1\}$
Çıktı: $x' \in \{0,\ \ldots,\ S-1\}$

1. $b \in_R \{0,1\}$ seç
2. $x \leq S-R$ ve $b = 1$ ise:
   $x' = x$
3. $x \leq S-R$ ve $b = 0$ ise:
   $x' = S-x$
4. $x > S-R$ ve $b = 1$ ise:
   $x' = x$
5. $x > S-R$ ve $b = 0$ ise:
   başa dön
6. x' çıkart

Şekil 4. Olasılıksal Eğilim Kaldırma Algoritması

4.2.2. Klepto RSA Anahtar Üretimi

Atağın yer aldığı çalışmada yazarlar, güvenliği Ayrık Logaritma Problemi'nin zorluğuna dayanan Diffie - Hellman anahtar değişimi [8] için de, kleptografik bir atak sunmuşlar ve daha sonra bu ataktaki stratejiyi RSA için Kleptografik anahtar üretim algoritmasında kullanmışlardır. Bu atağın detaylarına bu çalışmada girmeyeceğiz ancak [7]'de detaylı açıklama bulunabilir.
Atak algoritması saldırganın, $P \in \{0,1\}^k$ asal ve $g \in \mathbb{Z}_P^*$ üreteç eleman olmak üzere, Elgamal [4] (Y,g,P) açık anahtarı ve $Y = g^X \bmod P$ sağlayan X gizli anahtarı kullanmaktadır. P değeri ile kurbanın üretilecek p asalının bitsel gösterimleri eşit uzunluktadır. Yani $|P| = |p| = k$ olacaktır.
$G_K(a)$ fonksiyonu, K anahtarı ile a değerini rastgeleleştirmek için kullanacağımız bir G fonksiyonunu temsil edecektir. Bu fonksiyon simetrik şifreleme algoritmaları olabilir. Ancak; daha önce de belirttiğimiz gibi, tersi alınabilir olması gerektiğinden özet fonksiyonu kullanılamaz.

KleptoAnahtarÜreteç(k): [7]
Girdi: k, gizli asalların bit uzunluğu
Çıktı: Açık anahtar: (n,e) öyle ki $n \in \{\ \{0,1\}^{2k-1}, \{0,1\}^{2k}\}$, $1 < e < \varphi(n)$ ve $EBOB(e,\varphi(n)) = 1$
Gizli anahtar: d öyle ki $d \equiv e^{-1} \bmod \varphi(n)$

Rastgeleleştirme ve Özet fonksiyonları:
G: simetrik şifreleme fonksiyonları.(örn.: AES)
H: Kriptografik özet fonksiyonu
Gömülü Değerler:
Saldırgan ElGamal anahtarları: (Y,g,P) açık anahtar, X gizli anahtar
$P \in \{0,1\}^k$ asal, $g \in \mathbb{Z}_P^*$ üreteç, $X \in_R \mathbb{Z}_P^*$,
$Y = g^X \bmod P$
K anahtarı: G fonksiyonunda kullanılacak anahtar değeri.
$1 < W,a,b < P$ sabit tam sayılar.

1. $c_1 \in_R \{0,...,N-1\}$ seç.
2. $z \leftarrow g^{c_1 - Wt} Y^{-ac_1 - b} \bmod P$
3. $z' \leftarrow PBRM(P,2^k,z)$
4. $z'' = H(z')$
5. z' çift ise: $z' = z' + 1$
   z' tek ise: geç.
6. $i = 0$'dan B_1'e kadar:
   $p \leftarrow z'' + 2i$ (sadece tek olan değerleri deneniyor)
   p asal ise Adım 7'e geç.
   Değilse i'yi 1 arttır.
   Adım1'e dön.
7. $v \leftarrow \text{PBRM}(P,2^k,g^{c_1} \; mod N)$
8. $j = 0$'dan B_2'e kadar:
   $U \leftarrow G_{K+j}(v)$
   $RND \in_R \{0,1\}^k$ seç
   $n' \leftarrow (U \parallel RND)$
   $n' = pq + r$ den q'yu hesapla.
   q asalsa $n \leftarrow n' - r$ olarak belirle ve Adım 10'a geç.
   Değilse j'yi 1 arttır.
   Adım 1'e dön.
9. e ve d, RSA kuvvetlerini hesapla.

Şekil 5. Kleptografik RSA Anahtar Üretim Algoritması [7]

Adım 2'de aslında bir z mesajının ElGamal şifrelemeyle (r,s) şifreli metni elde edilmektedir. Daha sonra bu r ve s değerlerini birbirine eşitlenip z mesajı çekilmekte ve bu değerler atak algoritmasında kullanılmaktadır. Bu adımda yapılan atak Young ve Yung aynı çalışmada sundukları Ayrık Logaritma Atağı'nın temelini teşkil etmektedir. Burada ise Çarpanlara Ayırma Problemi'nin zorluğuna dayanan RSA şifreleme sisteminin gizli değerlerini çalmak için Ayrık Logaritma Problemi'nin zorluğundan faydalanılmaktadır.
Kullanıcının açık anahtarlarını ele geçiren saldırgan, aşağıdaki algoritmayla gizli anahtar değerine ulaşabilecektir.

AnahtarEleGeçirme (n,e,k): [7]
Girdi: (n,e) kurbanın açık anahtarları, k: gizli asalların bit uzunluğu
Saldırgan ElGamal anahtarları: (Y,g,P) açık anahtar, X gizli anahtar
$P \in \{0,1\}^k$ saldırganın ElGamal asalı, $g \in \mathbb{Z}_P^*$ üreteç,
Gizli Anahtar $X \in \mathbb{Z}_P^*$, Açık Anahtar $Y = g^X \bmod P$
K anahtarı: G fonksiyonunda kullanılan anahtar değeri.
$W,a,b \in \{0,\cdots,P\}$: saldırganın belirlediği sabit değerler,
$S = 2^k$: PBRM için üst sınır
Çıktı: d gizli anahtar

1. $U \leftarrow n^{\lceil |n|-k}$
2. $L_1 = \{v = G_{K+i}^{-1}(U) : i = 0,...,B_2;\}$
3. $L_2 = \{\text{PBRM}^{-1}(P,S,v) : v \in L_1\}$ ($g^{c_1} \bmod N$ için aday değerler)
4. $L_3 = \{z = hg^{-Wt}h^{-aX}Y^{-b} \bmod P :$
   $h \in L_2; t = 0,1\}$
5. $L_4 = \{z' : \forall z \in L_3$ için $z' = z$ ve $z' = S - z\}$
6. $L_5 = \{z'' : \forall z' \in L_4$ için $z'' =$
   $H(z')$ ve z'''nün en düşük biti 1 yapılır$\}$
7. $\forall z'' \in L_5$ için:
   $i \in \{0,...,B_1\}$ için:
   $p' = z'' + 2i$
   $p'|n$ ise:
   $p = p'$ yap ve bırak
   değilse $i = i + 1$
   p bulunamazsa başa dön ve $U = U + 1$ yap
8. $q = n/p$, $\varphi(n) = (p-1)(q-1)$
9. $d \equiv e^{-1} \bmod \varphi(n)$ çıkart

Şekil 6. Gizli Anahtar Ele Geçirme Algoritması [7]

Kullanıcının yayınladığı açık anahtarından gizli asalları elde etmek için yukarıdaki saldırı algoritmasını çalıştıran saldırgan, Adım 1'de p asalının bit sayısı olan k tane biti n modunun en düşük bitlerinden atmaktadır böylece rastgele eklenmiş bitleri çıkarır.
Adım 3'te PBRM metodunun tersini uygulayarak ($g^{c_1} \bmod P$) değerine ulaşır, sonraki adımda atak mekanizmasının şifrelediği bu değeri gizli anahtarını kullanarak ulaşır (burada şifreleyip/çözme işlemi yerine ortak z değerini bulabilme söz konusudur. Bu yüzden çalışmada ayrık logaritma atağı olarak nitelendirilmiştir). z değerini elde eden saldırgan bundan sonraki adımları aynen atak algoritmasının yaptığı gibi hesaplayarak p ve q değerlerine ulaşabilir.

4.2.3. Atağın Simülasyonu ve Analizi

Bölüm 4.1'de olduğu gibi bu atak için de, KleptoAnahtarÜreteç ve AnahtarEleGeçirme algoritmalarının gerçeklenmesi ve analizleri yapılmış ve sonuçlar bu bölümde ele alınacaktır. Bu bölümdeki atakla ilgili üretilen 150 adet anahtar için rastgele bir mesajı şifreleme/şifre çözme başarıları ve açık anahtardan, gizli anahtarı ele geçirebilme başarı oranları test edilmiş ve her iki durumun da %100 başarı ile sonuçlandığı belirlenmiştir. Algoritmaların gerçeklendiği bilgisayarın fiziksel durumu, işletim sistemi ve gerçeklendiği platform gibi detaylar Bölüm 1.4.1'da yer almaktadır.

- **Çalışma Zamanı:** Simüle ettiğimiz atak için üretilen 150 adet anahtar değerinin ve Bölüm RSA'de standart RSA anahtarları ile yapılan analizlerle karşılaştırmalı olarak, anahtar üretim zamanları, Tablo 5'de verilmiştir. Bu tabloda 512-bit anahtarlar için gereken çalışma zamanının yüksek çıkmasının sebebi, olması gerekenden (asallık testindeki işlemlerden) fazla olan moduler kuvvet alma işlemleri olarak görülebilir. Bu işlemler anahtar ele geçirirken de çalışma zamanlarını fazlaca artmasına (5-10 dk gibi) sebep olduğu görülmüştür.

Tablo 5. Çalışma Zamanı

asal.bit.uzunluğu:k	RSA	[7]
128	0,07	0,71
256	1,01	6,46
512	27,97	132,52

150 adet anahtarın üretim zamanı ortalamaları (sn.)

- **Açık Anahtarların Dağılımı:** Atak barındıran algoritmayla ürettiğimiz anahtarlar için, n açık anahtar değerinin, büyüklüğüne göre olasılık dağılımı, önceki atakta olduğu gibi standart RSA anahtarları ile karşılaştırmalı olarak Tablo 6'de verilmiştir. Bu tablodan da görülebileceği gibi, standart RSA anahtarları ile karşılaştırıldığında, atak barındıran sistemin ürettiği anahtar değerleri olması gereken dağılıma sahip değildir. Yani sistem, üretilen anahtarlarının dağılıma göre teste tabi tutulduğunda, sistemin atak barındırdığı ortaya çıkacaktır.

Tablo 6. Açık Anahtar Dağılım Tablosu

	RSA			[7]		
Bit Uzunluğu k	$2k-1$	$2k$ "10"	$2k$ "11"	$2k-1$	$2k$ "10"	$2k$ "11"
128	%38	%56	%6	%28	%36	%34
256	%36	%51	%12	%28	%38	%33
512	%37	%48	%14	%39	%28	%32

150 adet n açık anahtarın dağılım yüzdeleri

Sonuç olarak atak, Bölüm 4.1'deki atağa göre güvenliği arttırmış olsa da hala sistemin üreteceği çıktılara göre test edilmesi durumunda, standart algoritma ile ayrıt edilemezliği sağlayamamaktadır ve tespit edilebilir bir ataktır.

4.3. CS03: Crepeau ve Slakmon'un Coppersmith Teoremi'ni Kullandıkları RSA Kleptografik Atağı

Crepeau ve Slakmon [9] çalışmalarında, RSA şifreleme sistemi için toplamda 4 adet arka kapı kurgulamışlardır. Bu atakların 3'ünde p ve q asalları rastgele seçilmekte ve atak mekanizması açık ve gizli (e,d) kuvvet değerlerinin, bilinen bazı ataklara karşı zafiyet barındırabilecek şekilde üretilmektedir. Bu arka kapılardan sızdırılmak istenen değeri, sızdırmak için bir asimetrik şifreleme kullanılmamaktadır. Bu nedenle bu ataklar simetrik ataklar olarak nitelendirebiliriz ve kleptografi kapsamına girememektedir.
İkilinin çalışmalarındaki son ataklarında ise diğerlerinden farklı olarak kuvvet değerlerinin manipülasyonu ile değil ancak asallardan birinin bitsel gösteriminin üst yarısının (bitlerinin sol yarısı) sızdırılması şeklindedir. Çalışmadaki diğer 3 atak gibi bu atakta da sızdırılmak istenilen bilgiye asimetrik bir şifreleme uygulanmamakta ancak simetrik şifreleme olarak sınıflandırabileceğimiz bir şekilde permutasyona (rastgeleleştirme) tabi tutulmaktadır. Bu yüzden bu atağı da simetrik bir atak olarak nitelendiriyoruz. Atağın bu özelliğinden dolayı Kleptografi kapsamına girmese de *"Coppersmith Kısmi Bilgi Atağını"* kullanan ilk arka kapı çalışması olduğu için bu çalışmada yer almaktadır.

4.3.1. Coppersmith Kısmi Bilgi Atağı

Don Coppersmith [10] çalışmasında, aşağıdaki teoremi ispatlamıştır.

Teorem 4.1 (Coppersmith). *[10] $N = PQ$ çarpımı ve çarpanlardan birinin en üst $(1/4+\varepsilon)(\log_2 N)$ tane biti bilinirse, $\log N$ ve $1/\varepsilon$ a bağlı polinom zamanda N çarpımını, P ve Q çarpanlarına ayrılabilir.*

Bu teoreme göre RSA açık anahtarı N değeri, gizli asallardan birinin üst yarı bitleri bilindiği taktirde çarpanlarına ayrılabilmektedir. Bu durum kleptografik bir atakta, gizli bilgiyi sızdırmak için, asallardan birinin hepsini sızdırmak yerine üst yarı bitlerini sızdırmanın yeterli olacağı anlamına gelmektedir.

4.3.2. Permutasyon Fonksiyonu

Önceki ataklarda rastgeleleştirme ve döngüler oluşturabilmek için simetrik şifreleme fonksiyonlarını ve özet fonksiyonlarını kullanmıştık. Crepeau ve Salkmon [9] çalışmalarında permütasyon için kullandıkları fonksiyonları ele alırken, arka kapıların donanımsal uygulamasında, simetrik şifreleme fonksiyonlarının kullanılması durumunda, implementasyon alanının kapasitesinde problem oluşturabileceği için RSA ile aynı aritmetiğe sahip fonksiyonların kullanılmasının daha uygun olacağını belirtmişlerdir.
Yazarların, bu atak için önerdiği iki permutasyon, $\mu, \beta \in \mathbb{Z}^+$ sabit tamsayılar olmak üzere ve $|x|$ ifadesi x'in bitsel uzunluğunu ve $\mu\rfloor_n$ notasyonu ise μ'nun en düşük n tane bitini göstermek üzere, aşağıdaki gibidir;

$$\pi_{\beta,\mu}(x) = (x \oplus (2\mu)\rfloor_{|x|})^{-1} \bmod \beta$$

ve

$$\pi_{\beta,\mu}(x) = (x^{-1} \bmod \beta) \oplus (2\mu)\rfloor_{|\beta|}$$

Atağın implementasyonunu yaparken ikinci fonksiyonu kullanmayı tercih ettik. Bu fonksiyonun tersi aşağıdaki gibidir.

$$\pi^{-1}_{\beta,\mu}(x) = (x \oplus (2\mu)\rfloor_{|\beta|})^{-1} \bmod \beta$$

4.3.3. Atak

Arka kapı barındıran şifreleme sistemi aşağıdaki algoritmayla anahtar üretecektir. Algoritma sabitlenmiş e kuvvet değerine göre gerekli anahtarları üretebilecek şekilde sunulmuştur. $k-bit$ asal sayılar üreterek, çıktı olarak $2k$ veya $2k-1$ bit n açık anahtarını üretir.

KleptoAnahtarÜreteç(k,e): [9]
Girdi:
k: asal bit uzunluğu, e açık anahtar kuvvet değeri.
Çıktı:
Açık anahtar: (n,e) öyle ki $n \in \{\{0,1\}^{2k-1}, \{0,1\}^{2k}\}$, $1 < e < \varphi(n)$ ve $EBOB(e,\varphi(n)) = 1$
Gizli anahtar: d öyle ki $d \equiv e^{-1} \bmod \varphi(n)$
Gömülü Değerler:
$\beta, \mu \in_R \{0,1\}^{k/2}$ permütasyon fonksiyonunda kullanılacak sabit değerler.
Rastgelelştirme sonksiyonu:
$\pi_{\beta,\mu}(x) = (x^{-1} \bmod \beta) \oplus 2\mu\rfloor_{|\beta|}$
Operatörler:
$||$: bitsel dizileri uç uca ekleme operatörü.
$|x|$: x'in bitsel uzunluğu.
$\mu\rceil_n$: μ'nun en üst (sol) n tane biti.
$\mu\rfloor_n$: μ'nun en alt (sağ) n tane biti.

1. $p \in_R \{0,1\}^k$ ve $(e, p-1) = 1$ olacak şekilde p asalı seç
2. $q' \in_R \{0,1\}^k$ tek tamsayısını seç ve $n' = pq'$ hesapla
3. $n \leftarrow n'\rceil^{k/4} \,||\, \pi_{\beta,\mu}(p\rceil^{k/2}) \,||\, n'\rfloor_{5k/4}$
4. $q \leftarrow \lfloor n/p \rfloor$
5. q çift ise:
   $q = q+1$
6. $(e, q-1) > 1$ veya q asal değil iken:
   $m \in \{0,1\}^{k/4}$ ve çift m seç.
   $q \leftarrow q \oplus m$
7. $n \leftarrow pq$ ve $d \leftarrow e^{-1} \bmod \varphi(n)$ hesapla
8. (n,e) açık anahtarlar, (d) gizli anahtarları çıkart

Şekil 7. CS03 Kleptografik RSA Anahtar Üretim Algoritması

Yayınlanan n mod değerini elde eden saldırgan, Şekil 8'deki algoritma ile p ve q asallarına ulaşabilecektir.

Klepto anahtar üretim ve anahtar ele geçirme algoritmaları düzgün çalışacaktır. Çünkü; anahtar üretme algoritmasıyla üretilen n açık anahtar modül değeri

$$n \leftarrow n'\rceil^{k/4} || \pi_{\beta,\mu}(p\rceil^{k/2}) || n'\rfloor_{5k/4}$$

şeklinde belirlenmekte ve bu değerin sağ ve solunda birleştirilen parçalar gizli anahtarın ele geçirilmesiyle ilgili bir önem teşkil etmemektedir.

AnahtarEleGeçirme(n,β,μ): [9]
Girdi: (n,e) açık anahtarlar
$\beta, \mu \in \{0,1\}^{k/2}$ permütasyon fonksiyonunda kullanılan sabit değerler.
Çıktı: d gizli anahtar
Rastgeleleştirme fonksiyonu:
$\pi_{\beta,\mu}(x) = (x^{-1} \bmod \beta) \oplus 2\mu\rfloor_{|\beta|}$

1. $p_0 = \pi^{-1}_{\beta,\mu}(n\rceil^{3k/4}\rfloor_{k/2})$ hesapla $(p_0 = p\rceil^{k/2})$
2. $p, q = Coppersmith(n, p_0)$
3. $\varphi(n) = (p-1)(q-1)$
4. $d = e^{-1} \bmod \varphi(n)$ çıkart.

Şekil 8. CS03 Anahtar Ele Geçirme Algoritması

Saldırgan anahtarı ele geçirmek istediğinde bu parçaları atacak ve geriye kalan değer ise;

$$\pi^{-1}_{\beta,\mu}(\pi_{\beta,\mu}(p\rceil^{k/2})) = p\rceil^{k/2}$$

π^{-1} ters fonksiyonu ile beraber, gizli p asalının bitlerinin üst yarısını verecektir. Bundan sonra ise Coppersmith atağı ile

$$p, q = \text{Coppersmith}(n, p\rceil^{k/2})$$

hesaplayarak gizli asallara ulaşabilecektir.

4.3.4. Atağın Simülasyonu ve Analizi

Bölüm 4.1 ve 4.2'de olduğu gibi bu atak için de, KleptoAnahtarÜreteç ve AnahtarEleGeçirme algoritmalarının gerçeklenmesi ve analizleri yapılmış ve sonuçlar bu bölümde ele alınacaktır. Algoritmaların gerçeklendiği bilgisayarın fiziksel durumu, işletim sistemi ve gerçeklendiği platform gibi detaylar Bölüm 1.4.1'da yer almaktadır.
Diğer ataklarda olduğu gibi 150 adet anahtar değeri ile bu analizleri yapacağız. Bu bölümde ele aldığımız "Gizli Asal Çarpan" atağıyla ilgili üretilen 150 adet anahtar için rastgele bir mesajı şifreleme/şifre çözme başarıları ve açık anahtardan, gizli anahtarı ele geçirebilme başarı oranları test edilmiş. Ancak kurbanın açık anahtarından, gizli asalları ele geçirirken Coppersmith Kısmi Bilgi atağını kullanmak yerine, açık anahtardan ele geçirerek Coppersmith

algoritmasına girdi olarak verilmesi gereken, p asalının üst yarı bitlerini ele geçirebilme başarıları test edilmiştir. Yani Coppersmith Kısmi Bilgi atağının doğru çalışıyor olduğu kabulü altında gizli asallar ele geçirilebilir durumda olacaktır.
Bu şartlar altında şifreleme/şifre çözme başarıları ve açık anahtardan, gizli anahtarı ele geçirebilme başarı oranları %100 olarak gözlemlenmiştir.

- **Çalışma Zamanı:** Simüle ettiğimiz atak için üretilen 150 adet anahtar değerinin ve Bölüm RSA'de standart RSA anahtaları ile yapılan analizlerle karşılaştırmalı olarak, anahtar üretim zamanları, Tablo 7'de verilmiştir.

Tablo 7. Çalışma Zamanı

asal bit uzunluğu:k	RSA	CS03
128	0,07	0,20
256	1,01	3,35
512	27,97	48,03

150 adet anahtarın üretim zamanı ortalamaları (sn.)

- **Açık Anahtarın Dağılımı:** Atak barındıran algoritmayla ürettiğimiz anahtarlar için, n açık anahtar değerinin, büyüklüğüne göre olasılık dağılımı, önceki atakta olduğu gibi standart RSA anahtarları ile karşılaştırmalı olarak Tablo 8'de verilmiştir. Önceki bölümlerde ele aldığımız atakların aksine, bu atakta standart RSA anahtarları ile dağılım yüzdeleri uyuşmaktadır. Bu da atağın anahtarların dağılımını ölçerek tespit edilmesini engelleyecektir.

Tablo 8. Açık Anahtar Dağılım Tablosu

	RSA			CS03		
Bit Uzunluğu k	2k-1	2k "10"	2k "11"	2k-1	2k "10"	2k "11"
128	%42	%48	%9	%37	%52	%10
256	%34	%51	%14	%26	%58	%14
512	%38	%46	%15	%34	%60	%6

150 adet n açık anahtarın dağılım yüzdeleri

Sonuç olarak bu atak açık anahtarların ve anahtar üretimi çalışma zamanlarının standart RSA algoritmasıyla ile ayırt edilemezliği sağlamaktadır. Ancak daha önceden de bahsedildiği gibi atağın simetrik oluşu, atağı güvensiz hale getirmektedir.

5. Sonuç

Bu çalışmada RSA şifreleme sistemine karşı kurgulanmış kleptografik atak algoritmaları incelenmiş ve atakların simülasyonlarının istatistiksel sonuçları ölçülmüştür. Yer kısıtından dolayı ele alamadığımız ancak istatistiksel testlerde başarılı sonuçların gözlemlendiği ve aynı zamanda güvenliğin de sağlanabildiği başka çalışmalar da mevcuttur [12–14]. Bu çalışmalarda eliptik eğriler kullanılmakta ve detaylı analizler ve test sonuçları [33] çalışmasında bulunabilir.

Sonuç olarak istatistiksel testlerden geçebilecek kleptografik arka kapı uygulamalarının mümkün olduğu görülmüştür. Bu sonuçlar ışığında; kapalı kutu yani çalışan kodun görülüp analiz edilemediği sistemlerin arka kapı barındıryor olma risklerinin olduğunu ve böyle sistemlerden mümkün olduğunca kaçınılması gerektiğini söyleyebiliriz.

Teşekkür

Bu çalışmada 1. yazar, TÜBİTAK tarafından "2211 Yurt İçi Lisansüstü Burs Programı" kapsamında desteklenmiştir. 3. yazar ise "(114C027) funded by EU FP7-The Marie Curie Action and Tübitak (2236-CO-FUNDED Brain Circulation Scheme)" projesinden desteklenmiştir. Desteklerinden ötürü ilgili kurumlara şükranlarımızı sunuyoruz.

Kaynakça

[1] A. Young and M. Yung, "The dark side of "black-box" cryptography or: Should we trust capstone?" in *Advances in Cryptology—CRYPTO'96*. Springer, 1996, pp. 89–103.

[2] G. J. Simmons, "The subliminal channel and digital signatures," in *Advances in Cryptology*. Springer, 1984, pp. 364–378.

[3] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[4] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Advances in cryptology*. Springer, 1984, pp. 10–18.

[5] P. FIPS, "186-2. Digital Signature Standard (DSS)," *National Institute of Standards and Technology (NIST)*, 2000.

[6] B. C. Neuman and T. Ts' O, "Kerberos: An authentication service for computer networks," *Communications Magazine, IEEE*, vol. 32, no. 9, pp. 33–38, 1994.

[7] A. Young and M. Yung, "Kleptography: Using cryptography against cryptography," in *Advances in Cryptology—Eurocrypt'97*. Springer, 1997, pp. 62–74.

[8] W. Diffie and M. E. Hellman, "New directions in cryptography," *Information Theory, IEEE Transactions on*, vol. 22, no. 6, pp. 644–654, 1976.

[9] C. Crépeau and A. Slakmon, "Simple backdoors for RSA key generation," in *Topics in Cryptology—CT-RSA 2003*. Springer, 2003, pp. 403–416.

[10] D. Coppersmith, "Finding a small root of a bivariate integer equation; factoring with high bits known," in *Advances in cryptology—EUROCRYPT'96*. Springer, 1996, pp. 178–189.

[11] A. Young and M. Yung, *Malicious cryptography: Exposing cryptovirology*. John Wiley & Sons, 2004.

[12] A. L. Young and M. Yung, "A space efficient backdoor in RSA and its applications," in *Selected Areas in Cryptography*. Springer, 2006, pp. 128–143.

[13] A. L. Young and M. Yung, "Space-efficient kleptography without random oracles," in *Information Hiding*. Springer, 2007, pp. 112–129.

[14] A. Young and M. Yung, "Kleptography from standard assumptions and applications," in *Security and Cryptography for Networks*. Springer, 2010, pp. 271–290.

[15] Z. Golebiewski, M. Kutyłowski, and F. Zagórski, "Stealing secrets with ssl/tls and ssh–kleptographic attacks," in *Cryptology and Network Security*. Springer, 2006, pp. 191–202.

[16] E. J. Goh, D. Boneh, B. Pinkas, and P. Golle, "The design and implementation of protocol-based hidden key recovery," in *Information Security*. Springer, 2003, pp. 165–179.

[17] M. Gogolewski, M. Klonowski, P. Kubiak, M. Kutyłowski, A. Lauks, and F. Zagórski, "Kleptographic attacks on e-voting schemes," in *Emerging Trends in Information and Communication Security*. Springer, 2006, pp. 494–508.

[18] M. Gogolewski, M. Gomu^{a}kiewicz, J. Kubiak, and M. Lauks, "Kleptographic attaks on e-auction schemes," *Tatra Mt. Math. Publ*, vol. 41, no. 47, pp. 47–64, 2008.

[19] N. Perlroth, J. Larson, and S. Shane, "NSA able to foil basic safeguards of privacy on web," *The New York Times*, vol. 5, 2013.

[20] J. Ball, J. Borger, and G. Greenwald, "Revealed: how US and UK spy agencies defeat internet privacy and security," *The Guardian*, vol. 6, 2013.

[21] E. B. Barker and J. M. Kelsey, *Recommendation for random number generation using deterministic random bit generators (revised)*. US Department of Commerce, Technology Administration, National Institute of Standards and Technology, Computer Security Division, Information Technology Laboratory, 2007.

[22] S. Checkoway, R. Niederhagen, A. Everspaugh, M. Green, T. Lange, T. Ristenpart, D. J. Bernstein, J. Maskiewicz, H. Shacham, and M. Fredrikson, "On the practical exploitability of dual ec in tls implementations," in *23rd USENIX Security Symposium (USENIX Security 14)*, 2014, pp. 319–335.

[23] K. G., "Dual-EC-PRBG Comments," http://www.math.ntnu.no/~kristiag/drafts/dual-ec-drbg-comments.pdf, 2006, (Son Erişim: Haziran 2016).

[24] D. S. and N. F., "On the possibility of a back door in the nist sp800-90 Dual-EC-PRNG. crypto 2007 rump session,," http://rump2007.cr.yp.to/15-shumow.pdf., 2007., (Son Erişim: Haziran 2016).

[25] B. Schoenmakers and A. Sidorenko, "Cryptanalysis of the dual elliptic curve pseudorandom generator." *IACR Cryptology ePrint Archive*, vol. 2006, p. 190, 2006.

[26] I. Mironov and N. Stephens-Davidowitz, "Cryptographic reverse firewalls," in *Advances in Cryptology-EUROCRYPT 2015*. Springer, 2015, pp. 657–686.

[27] A. Russell, Q. Tang, M. Yung, and H. S. Zhou, "Cliptography: Clipping the power of kleptographic attacks," Cryptology ePrint Archive, Report 2015/695, 2015. http://eprint. iacr. org, Tech. Rep., 2015.

[28] K. Ruohonen, "Mathematical cryptology," *Lecture Notes*, 2010.

[29] J. Hoffstein, J. Pipher, J. H. Silverman, and J. H. Silverman, *An introduction to mathematical cryptography*. Springer, 2008, vol. 1.

[30] E. Güzel. Erhan Güzel Cebir Sayfası http://web.iku.edu.tr/~eguzel (Son Erişim: Haziran 2016).

[31] Marmara Üniversitesi Fen-Edebiyat Fakültesi Cebir Ders Notları http://mat.fef.marmara.edu.tr/ogrencilere/cebir-ii-ders-notlari/ (Son Erişim: Haziran 2016).

[32] D. Boneh *et al.*, "Twenty years of attacks on the RSA cryptosystem," *Notices of the AMS*, vol. 46, no. 2, pp. 203–213, 1999.

[33] E. Ceran, M.S. Kiraz, O. Uzunkol, 2016. Kleptografi: Kriptografik Sistemlerde Arka Kapılar. İstanbul Şehir Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, 71s, İstanbul.