T.C.
ERZİNCAN BİNALİ YILDIRIM ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MATEMATİK ANABİLİM DALI

DİJİTAL İMZA ALGORİTMALARININ MATEMATİKSEL İNCELENMESİ

ALEYNA GÖGEN

Danışman: Dr. Öğr. Üyesi İsrafil OKUMUŞ

TEZ JÜRİ ÜYELERİ
Prof. Dr. Mustafa KUDU
Doç. Dr. Muhammed YİĞİDER
Dr. Öğr Üyesi İsrafil OKUMUŞ

YÜKSEK LİSANS TEZİ
ERZİNCAN, 2025

Kabul ve Onay Sayfası

Dr. Öğr. Üyesi İsrafil OKUMUŞ danışmanlığında, Aleyna GÖGEN tarafından hazırlanan bu çalışma 19/11/2025 tarihinde aşağıdaki jüri tarafından Matematik Anabilim Dalı'nda Yüksek Lisans Tezi olarak oybirliği (3/3) ile kabul edilmiştir.

Başkan:	Prof. Dr. Mustafa KUDU	İmza:
Üye:	Doç. Dr. Muhammed YİĞİDER	İmza:
Üye:	Dr. Öğr. Üyesi İsrafil OKUMUŞ	İmza:

Yukarıdaki Yüksek Lisans Tezi Enstitü Yönetim Kurulunun …. / …. / 20.... tarih ve …../…….. sayılı kararı ile onaylanmıştır.

Doç. Dr. Kemal Volkan ÖZDOKUR

Enstitü Müdür V.

Not: Bu tezde kullanılan özgün ve başka kaynaklardan yapılan bildirişlerin, şekil ve tabloların kaynak olarak kullanımı, 5846 sayılı Fikir ve Sanat Eserleri Kanunundaki hükümlere tabidir.

Bilimsel Etiğe Uygunluk Sayfası

"Dijital İmza Algoritmalarının Matematiksel İncelenmesi" isimli "Yüksek Lisans" tezim tarafımca intihal tespit programı ile incelenmiştir. Buna göre tezimde bilimsel etik ihlali ve intihal olarak nitelendirilebilecek herhangi bir durum olmadığını taahhüt ederim.

Bu çalışmadaki tüm bilgilerin, akademik ve etik kurallara uygun bir biçimde elde edildiğini; aynı zamanda bu kural ve davranışların gerektirdiği gibi, bu çalışmanın özünde olmayan tüm materyal ve sonuçları tam olarak aktardığımı ve referans gösterdiğimi beyan ederim. 19/11/2025

(İmza)

Aleyna GÖGEN

ÖZET

DİJİTAL İMZA ALGORİTMALARININ MATEMATİKSEL İNCELENMESİ

Aleyna GÖGEN

Yüksek Lisans Tezi, Erzincan Binali Yıldırım Üniversitesi, Fen Bilimleri Enstitüsü, Matematik Anabilim Dalı
Danışman: Dr. Öğr. Üyesi İsrafil OKUMUŞ

2025, 76 sayfa

Bu tezde, klasik dijital imza algoritmaları, kör imza algoritmaları ve hash tabanlı temel kuantum dayanıklı imza algoritmaları kapsamlı bir biçimde incelenmiştir. İlk olarak, dijital imza mekanizmalarının matematiksel temelini oluşturan temel kavramlar tanıtılmış; bu kavramlara dayanan RSA, Rabin ve ElGamal gibi asimetrik şifreleme algoritmaları açıklanmıştır. Devamında, klasik dijital imza şemalarından RSA, Rabin, ElGamal, Fiat–Shamir, Guillou–Quisquater, Schnorr, DSA ve Nyberg–Rueppel algoritmaları ile bunların eliptik eğri tabanlı türevleri ele alınmış; ayrıca RSA, Rabin, Schnorr, DSA ve Nyberg–Rueppel algoritmalarının kör imza versiyonları ayrıntılı olarak incelenmiştir.

Kuantum bilgisayarların klasik kriptografik sistemler üzerinde oluşturduğu tehditler çerçevesinde, hash tabanlı kuantum dayanıklı imza algoritmalarından Lamport ve Winternitz algoritmaları ele alınmıştır. Buna ek olarak, inkâr edilemez imza modeli kapsamında Chaum–van Antwerpen imza algoritması ile saldırı tespit edildiğinde imzalama işlemini durdurabilen GMR fail-stop imza algoritması incelenmiştir.

Tez boyunca, her bir imza algoritmasının cebirsel işleyiş adımları ayrıntılı olarak sunulmuş ve geçerlilikleri özgün ispatlarla ortaya konulmuştur. Ayrıca, algoritmaların olasılıksal ve deterministik özellikleri ile güvenlik varsayımları analiz edilerek anlaşılabilirliği artırmak amacıyla her bir algoritma için örneklere yer verilmiştir.

Anahtar Kelimeler: Açık anahtarlı kriptografi, Dijital imza, Kör imza.

ABSTRACT

MATHEMATİCAL ANALYSİS OF DIGITAL SİGNATURE ALGORİTHMS

Aleyna GÖGEN

Master's Thesis, Erzincan Binali Yıldırım University, Institute of Science and Technology,
Department of Mathematics
Advisor: Asst. Prof. Dr. İsrafil OKUMUŞ

2025, 76 pages

In this thesis, classical digital signature algorithms, blind signature schemes, and basic hash-based quantum-resistant signature algorithms are comprehensively investigated. First, the fundamental concepts forming the mathematical basis of digital signature mechanisms are introduced, and asymmetric cryptographic algorithms such as RSA, Rabin, and ElGamal, which are built upon these concepts, are explained. Subsequently, classical digital signature schemes including RSA, Rabin, ElGamal, Fiat–Shamir, Guillou–Quisquater, Schnorr, DSA, and Nyberg–Rueppel, along with their elliptic curve–based variants, are examined. In addition, the blind signature versions of the RSA, Rabin, Schnorr, DSA, and Nyberg–Rueppel algorithms are analyzed in detail. Within the framework of the threats posed by quantum computers to classical cryptographic systems, fundamental hash-based quantum-resistant signature algorithms such as the Lamport and Winternitz schemes are studied. Furthermore, the Chaum–van Antwerpen signature algorithm, which operates under the undeniable signature model, and the GMR fail-stop signature algorithm, which can halt the signing process upon attack detection, are also investigated. Throughout the thesis, the algebraic operational steps of each signature algorithm are presented in detail, and their correctness is rigorously proven through original derivations. Moreover, the probabilistic and deterministic properties of the algorithms, together with their underlying security assumptions, are analyzed, and illustrative examples are provided for each algorithm to enhance clarity.

Keywords: Public-key cryptography, Digital signature, Blind signature.

TEŞEKKÜR

Yüksek lisans eğitimim süresince bilgi, birikim ve desteğiyle her zaman yanımda olan, tezim boyunca bana rehberlik eden değerli danışmanım Dr. Öğr. Üyesi İsrafil OKUMUŞ'a en içten teşekkürlerimi sunarım.

Akademik hayatım boyunca beni her zaman destekleyen, sevgi ve anlayışlarını esirgemeyen babam Emrullah GÖGEN'e, annem Züleyha GÖGEN'e ve kardeşlerime sonsuz teşekkürlerimi sunarım.

Aleyna GÖGEN
Kasım, 2025

İÇİNDEKİLER

TABLOLAR DİZİNİ

ŞEKİLLER DİZİNİ

SİMGELER VE KISALTMALAR DİZİNİ

AES	Advanced Encryption Standard
DES	Data Encryption Standard
DSA	Digital Signature Algortihm
ECC	Elliptic Curve Cryptography
IBM	International Business Machines
MD4	Message-Digest Algorithm 4
NIST	National Institue of Standards and Technology
NSA	National Security Agency
RSA	Rivest, Shamir and Adleman
SHA1	Secure Hash Algorithm-1
φ	Euler Phi Fonksiyonu
Z_n	$\{0,1,2,\dots,n-1\}$
Z_n^*	$\{a \in Z_n : (a,n) = 1\}$
(a,b)	EBOB
$[a,b]$	EKOK
$H(m)$	Hash (Özet) Fonksiyonu
$\{0,1\}^n$	n-bit uzunluğunda sayı
$a \parallel b$	Birleştirme (concatenation)

1. GİRİŞ

Bilgi ve iletişim teknolojilerinin hızla gelişmesiyle birlikte, elektronik ortamda yürütülen işlemlerin güvenliğinin ve geçerliliğinin sağlanması, çağdaş toplumların temel gereksinimlerinden biri hâline gelmiştir. Bu doğrultuda dijital imza sistemleri; veri bütünlüğü, kimlik doğrulama, inkâr edilemezlik ve gizlilik gibi temel güvenlik ilkelerini sağlayarak dijital güvenlik altyapılarının vazgeçilmez bir unsurunu oluşturmaktadır. Dijital imzalar, elle atılan imzalarla hukuken eşdeğer kabul edilmekte ve resmi işlemlerin kâğıt ortamına kıyasla çok daha hızlı, güvenli ve düşük maliyetle elektronik ortamda gerçekleştirilmesine imkân tanımaktadır.

Dijital imzalar temelde asimetrik kriptografi prensibine dayanır ve özel–genel anahtar çiftleri kullanılarak çalışır. İmzalama işlemi özel anahtar ile, doğrulama işlemi ise genel anahtar ile gerçekleştirilir. Bu çerçevede geliştirilen dijital imza algoritmaları, kriptografi tarihinde önemli bir dönüm noktası niteliği taşımaktadır. Klasik dijital imza sistemlerinin güvenliği; asal sayılar teorisi, modüler aritmetik, sonlu cisimler ve eliptik eğri cebiri gibi matematiğin çeşitli alanlarına dayanmaktadır. Bu algoritmaların güvenliği, klasik bilgisayar ile hesaplanması pratik olarak mümkün olmayan zorlu matematiksel problemlere bağlıdır. Örneğin RSA algoritması asal çarpanlara ayırma probleminin; ElGamal ve Schnorr algoritmaları ise ayrık logaritma probleminin hesaplanamazlığına ilişkin varsayımlara dayalı güvenlik sunar. Eliptik Eğri Kriptografisi (ECC), daha küçük anahtar boyutlarıyla yüksek güvenlik seviyeleri üretmesi sayesinde özellikle kaynak kısıtlı ortamlarda önemli bir verimlilik avantajı sağlamaktadır.

Bununla birlikte, bu algoritmaların büyük bölümü kuantum bilgisayarların varlığı altında güvenliğini yitirmektedir. Kuantum dayanıklı imza şemaları olan Lamport, Winternitz, XMSS, LMS ve SPHINCS+ hash fonksiyonlarının tek-yönlülüğüne dayanırken; CRYSTALS-Dilithium ve Falcon gibi ızgara (lattice) tabanlı imza algoritmaları ise en kısa vektör (SVP) ve en yakın vektör (CVP) problemlerinin hesaplamaya karşı dayanıklılığını temel almaktadır.

1.1. Araştırmanın Amacı

Bu tezin amacı, dijital imza algoritmalarının matematiksel temellerini, çalışma prensiplerini ve güvenlik özelliklerini bütüncül bir yaklaşımla incelemektir. Bu doğrultuda, klasik dijital imza şemaları ile kuantum sonrası döneme uyumlu hash-tabanlı temel imza algoritmaları ele alınmış; bu algoritmaların dayandığı teorik yapılar, cebirsel işleyiş mekanizmaları ve güvenlik varsayımları değerlendirilmiştir.

Tez kapsamında her bir algoritmanın cebirsel adımları verilmiş, doğruluk ispatları gösterilmiş ve somut örnekler verilerek bu imza sistemlerinin matematiksel boyutunun daha iyi anlaşılması hedeflenmiştir.

1.2. Araştırmanın Önemi

Dijital imza sistemleri, günümüzde dijital iletişim, e-ticaret, kurumsal bilgi yönetimi ve kimlik doğrulama süreçlerinin temel bileşenlerinden biri hâline gelmiştir. Bu sistemlerin güvenilirliği ise doğrudan kullanılan imza algoritmalarının matematiksel temellerine ve kriptografik dayanıklılığına bağlıdır. Klasik dijital imza algoritmaları uzun yıllar boyunca güvenli kabul edilse de, kuantum bilgisayarların ortaya çıkışı mevcut şifreleme altyapılarının gelecekte zayıflayabileceğine dair güçlü bir risk oluşturmaktadır.

Bu nedenle, hem hâlihazırda kullanılan imza şemalarının matematiksel sınırlarının ve güvenlik varsayımlarının kapsamlı biçimde anlaşılması hem de kuantum dayanıklı alternatif imza algoritmalarının teorik temelleriyle birlikte incelenmesi kritik bir gereklilik hâlini almıştır. Bu tez, söz konusu boşluğu doldurarak dijital imza teknolojilerinin mevcut durumuna ve gelecekteki kriptografik dönüşümlere bilimsel ve bütüncül bir bakış açısı sunmayı amaçlamaktadır.

1.3. Varsayımlar

Bu çalışmada, incelenen kriptografik algoritmaların güvenliği; çözümünün mevcut hesaplama gücüyle pratikte mümkün olmadığı kabul edilen matematiksel problemlere dayandığı sürece geçerli sayılmıştır. Bu kapsamda, klasik imza algoritmalarında asal çarpanlara ayırma ve ayrık logaritma gibi problemlerin çözülemezliği; kuantum dayanıklı imza şemalarında ise hash fonksiyonlarının tek-yönlülüğü temel güvenlik varsayımı olarak ele alınmıştır.

1.4. Sınırlılıklar

Bu çalışmada, kapsam sınırlı tutulmuş ve klasik dijital imza algoritmaları ile Lamport ve Winternitz gibi kuantum dayanıklı hash-tabanlı temel imza şemaları ele alınmıştır. Izgara (lattice) tabanlı kuantum dayanıklı imza algoritmaları ile güncel kuantum dayanıklı imza standartları bu tez kapsamında incelenmemiştir. Ayrıca yazılımsal implementasyonlar, performans karşılaştırmaları, donanımsal uygulamalar ve gerçek zamanlı testler çalışmanın dışında bırakılmıştır.

Bu tezde incelenen dijital imza algoritmalarının kronolojik sıralaması aşağıda sunulmuştur:

- RSA İmza Algoritması (1978)
- Rabin İmza Algoritması (1979)
- Lamport Kuantum Dayanıklı Tek Kullanımlık İmza Algoritması (1979)
- Merkle–Lamport Kuantum Dayanıklı Tek Kullanımlık İmza Algoritması (1979)
- Winternitz Kuantum Dayanıklı Tek Kullanımlık İmza Algoritması (1982)
- RSA Kör İmza Algoritması (1983)
- ElGamal İmza Algoritması (1984)
- Guillou–Quisquater İmza Algoritması (1985)
- Fiege–Fiat–Shamir İmza Algoritması (1986)
- Schnorr İmza Algoritması (1989)
- Chaum–van Antwerpen İnkar Edilemez İmza Algoritması (1989)
- Dijital İmza Algoritması – DSA (1991)
- Nyberg–Rueppel İmza Algoritması (1993)
- Nyberg–Rueppel Kör İmza Algoritması (1993)
- GMR Fail-Stop İmza Algoritması (1993)

Bu sınırlar çerçevesinde çalışma, dijital imza şemalarının tarihsel gelişimini, matematiksel temellerini ve kuantum öncesi-kuantum sonrası döneme ilişkin temel yaklaşımları teorik bir perspektifle incelemeye odaklanmıştır.

Dijital imza literatürü, 1970'li yılların sonlarında asimetrik kriptografinin gelişmesiyle şekillenmeye başlamıştır. Bu dönemde önerilen RSA İmza Algoritması (1978), Rivest, Shamir ve Adleman tarafından geliştirilmiş olup güvenliğini büyük tam sayıların çarpanlara ayrılmasının zorluğuna dayandırmaktadır. RSA'yı izleyen Rabin İmza Algoritması (1979) ise birleşik mod altında karekök alma problemini temel alarak geliştirilmiştir.

Kuantum öncesi dönemde hash tabanlı imza yapılarının temelleri Lamport Tek Kullanımlık İmza Algoritması (1979) ve Merkle-Lamport OTS (1979) ile atılmıştır. Bu iki yapı, hash fonksiyonlarının tek yönlülüğünden yararlanarak güvenlik sağlayan ve modern kuantum dayanıklı imza şemalarının öncülleri kabul edilen algoritmalardır. Bu çizgiyi takip eden Winternitz Tek Kullanımlık İmza Algoritması (1982) ise hash tabanlı imzalarda verimliliği artıran önemli bir iyileştirme olarak literatüre girmiştir.

Dijital imza literatüründe gizlilik ve anonimlik kavramlarını başlatan ilk çalışma, Chaum'un RSA Kör İmza Algoritması (1982) olmuştur. Chaum'un önerdiği kör imza modeli, imzalayanın mesaj içeriğini öğrenemediği güvenlik yapısı sayesinde e-oylama ve anonim kimlik doğrulama sistemlerinin temellerini oluşturmuştur.

Asimetrik imzalar alanındaki diğer önemli katkı ElGamal İmza Algoritması (1984) olup güvenliğini ayrık logaritma probleminin çözüm zorluğuna dayandırmaktadır. ElGamal yapısı, daha sonra geliştirilecek pek çok modern imza algoritmasına altyapı oluşturmuştur. Aynı dönemde kimlik doğrulamaya dayalı protokoller, imza sistemlerine uyarlanmış; Guillou-Quisquater (1985) ve Fiege-Fiat-Shamir (1986) algoritmaları sıfır bilgi ispatlarının dijital imza yapılarıyla bütünleştirilmesine öncülük etmiştir.

Ayrık logaritmaya dayalı Schnorr İmza Algoritması (1989), düşük imza boyutu ve yüksek verimliliği ile dikkat çekmiş ve günümüzde birçok modern şemanın matematiksel temelini oluşturan algoritmalardan biri hâline gelmiştir. Schnorr'un 1990 tarihli çalışması, imzanın verimliliğini artıran ek teknik iyileştirmeler içermektedir.

İnkar edilemez imza kavramı Chaum-Van Antwerpen (1989) tarafından tanıtılarak literatüre farklı bir doğrulama modeli kazandırmıştır. Devamında ABD tarafından standart hâline getirilen Dijital İmza Algoritması - DSA (1991) dünya genelinde yaygın bir kullanım alanı bulmuştur.

1990'lı yılların başında geliştirilen imza şemaları arasında Nyberg-Rueppel (1993) ve Nyberg-Rueppel Kör İmza Algoritması (1993) yer almakta olup her iki algoritma da klasik ve kör imza yapılarında verimlilik ve güvenlik açısından önemli katkılar sunmuştur. Aynı yıl önerilen GMR Fail-Stop İmza Algoritması (1993) ise saldırı tespiti hâlinde imzalama işlemini durdurabilme özelliği ile fail-stop güvenlik modelini literatüre kazandırmıştır. 1980'li yılların sonlarında geliştirilen Eliptik Eğri Kriptografisi (ECC), daha küçük anahtar boyutları ile

yüksek güvenlik seviyeleri sunması nedeniyle literatürde önemli bir atılım olarak yer almıştır (Koblitz, 1987; Miller, 1986). ECC tabanlı imza şemaları özellikle mobil ve gömülü sistemler gibi kaynak kısıtlı ortamlarda yaygın şekilde tercih edilmektedir.

Son yıllarda kuantum hesaplama teknolojilerinin hızla ilerlemesi klasik dijital imza algoritmalarının uzun vadeli güvenliğini tehdit etmektedir. Bu nedenle literatürde hash tabanlı, lattice tabanlı ve multivariate polinom temelli kuantum dayanıklı imza algoritmalarına yönelik çalışmalar yoğunlaşmıştır. NIST tarafından yürütülen Post-Quantum Cryptography standardizasyon süreci (NIST, 2022), söz konusu algoritmaların güvenlik analizlerini ve performans değerlendirmelerini kapsamaktadır. Bu literatür çizgisi, dijital imza algoritmalarının tarihsel gelişimini, matematiksel temellerini ve kuantum sonrası döneme uzanan güvenlik yaklaşımlarını bütüncül bir perspektifle ortaya koymaktadır.

Dijital imza ve algoritmaları üzerine ülkemizde Matematik, Elektrik-Elektronik Mühendisliği, Bilgisayar Mühendisliği ve diğer bilim dallarında yapılmış lisansüstü çalışmaların sayıları YÖK TEZ veri tabanından ilgili anahtar kelimeler kullanılarak araştırılmış olup sonuçlar Tablo 1 de verilmiştir.

Tablo 1. Dijital imza ile ilgili yapılmış lisansüstü çalışmalar

Bilim Dalı	Yüksek Lisans	Doktora
Hukuk	42	7
Matematik	10	9
Elektrik-Elektronik Mühendisliği	20	3
Bilgisayar Mühendisliği	79	16
İşletme	12	3
Bilim ve Teknoloji	9	3
Bilgi ve Belge Yönetimi	11	4
Adli Tıp	3	2
Endüstri ve Endüstri Mühendisliği	2	1

Erişim Tarihi: 23.05.2025

Bu tez çalışması beş bölümden oluşmaktadır. İlk bölüm olan Giriş'in ardından gelen Kavramsal Çerçeve bölümünde, tez kapsamında kullanılan temel tanımlar, teoremler ve matematiksel altyapı sunulmuştur. Yöntem bölümünde ise terminoloji, şifreleme ve dijital imza şemaları, klasik şifreleme algoritmaları ve eliptik eğri tabanlı şifreleme yöntemleri ele alınmıştır. Bulgular bölümünde klasik dijital imza algoritmaları ile temel kuantum dayanıklı dijital imza şemaları incelenmiştir ve matematiksel ve güvenlik açısından analizleri

yapılmıştır. Son bölüm olan Tartışma ve Sonuç bölümünde ise incelenen algoritmaların genel değerlendirmelerine yer verilmiştir.

2. KAVRAMSAL ÇERÇEVE

Bu bölümde, dijital imza algoritmalarının matematiksel altyapısını oluşturan temel kavramlar sunulmaktadır. İlk olarak sayı kuramına ilişkin bölünebilme, asal sayılar ve kongrüanslar gibi temel yapılar ele alınmış; ardından soyut cebirin grup, halka ve cisim gibi temel kavramları açıklanmıştır. Son olarak eliptik eğrilerin cebirsel özellikleri, nokta işlemleri ve sonlu cisimler üzerindeki tanımları incelenerek, ilerleyen bölümlerde yer alan imza algoritmalarının anlaşılmasına gerekli teorik temel oluşturulmuştur.

2.1. Sayılar Kuramı

Bu bölümdeki temel tanım ve teoremler (Altındiş, 2005); (Asar, 2012) ve (Rosen 2015) referanslı kaynaklardan derlenmiştir.

2.1.1. Bölünebilme

Tanım 2.1. $a \neq 0$ ve b tam sayılar olmak üzere $b = a.t$ olacak şekilde bir t tam sayısı varsa $a,\ b$'yi (tam) böler denir ve $a|b$ şeklinde gösterilir.

Teorem 2.1. a, b ve c tam sayılar olmak üzere aşağıdaki özellikler sağlanır.

i. $a|b$ ise $a|bc$.
ii. $a|b$ ise $ac|bc$.
iii. $a|b$ ve $b|c$ ise $a|c$.
iv. $a|b$ ve $a|c$ ise her $x, y \in \mathbb{Z}$ için $a|(x \cdot b + y \cdot c)$.

Teorem 2.2. (Bölme algoritması) $a > 0$ ve b tam sayılar olmak üzere

$$b = q \cdot a + r \text{ ve } 0 \leq r < a$$

olacak şekilde q ve r tam sayıları tek türlü olarak belirlenir.

Örnek 2.1. 20 sayısının ikilik tabandaki karşılığı bölme algoritması yardımı ile aşağıdaki şekilde hesaplanır:

$$\begin{aligned} 20 &= 2.10 + 0 \\ 10 &= 2.5 + 0 \\ 5 &= 2.2 + 1 \end{aligned}$$

$$2 = 2.1 + 0$$
$$1 = 2.0 + 1$$

Bu durumda 20 sayısı ikilik tabanda $20 = (10100)_2$ olarak yazılır. Bilgisayar terminolojisinde bu ifade 5 bit uzunluğunda bir sayı olarak adlandırılır.

Tanım 2.2. (En büyük ortak bölen) $a_1, a_2, \cdots, a_n$ en az biri sıfır olmayan tam sayılar olmak üzere $d|a_1, d|a_2, \cdots, d|a_n$ olacak şekildeki $d > 0$ sayısına $a_1, a_2, \cdots, a_n$ tam sayılarının ortak böleni denir. Bu $a_1, a_2, \cdots, a_n$ tam sayılarının ortak bölenlerinin en büyüğü $(a_1, a_2, \cdots, a_n)$ ile gösterilir. $(a_1, a_2, \cdots, a_n) = 1$ ise bu sayılara aralarında asaldır denir.

Algoritma 2.1. (Öklid algoritması) $a > 0$ ve b tam sayılar olmak üzere bölme algoritması $r_s = 0$ oluncaya kadar art arda uygulanarak aşağıdaki eşitlikler hesaplanır ve $(a, b) = r_{s-1}$ olarak elde edilir.

$$b = q_0 a + r_0, \qquad 0 \le r_0 < a$$
$$a = q_1 r_0 + r_1, \qquad 0 \le r_1 < r_0$$
$$r_0 = q_2 r_1 + r_2, \qquad 0 \le r_2 < r_1$$
$$r_1 = q_3 r_2 + r_3, \qquad 0 \le r_3 < r_2$$
$$\vdots \qquad\qquad \vdots$$

Ayrıca yukarıdaki eşitliklerden sırasıyla $r_{s-2}, r_{s-3}, \cdots, r_0$ yok edilerek $r_{s-1} = a \cdot x_0 + b \cdot y_0$ olacak biçimde x_0, y_0 tam sayıları bulunabilir. Bu işlem Genişletilmiş Öklid Algoritması olarak adlandırılır.

Örnek 2.2. $(35,43) = 35x + 43y$ eşitliğini sağlana x ve y tam sayıları Genişletilmiş Öklid Algoritması yardımı ile aşağıdaki şekilde edilir.

$$43 = 35 \cdot 1 + 8 \Longrightarrow 8 = 43 - 35.1$$
$$35 = 8 \cdot 4 + 3 \Longrightarrow 3 = 35 - 8.4$$
$$8 = 3 \cdot 2 + 2 \Longrightarrow 2 = 8 - 3.2$$
$$3 = 2 \cdot 1 + 1 \Longrightarrow 1 = 3 - 2 \cdot 1$$
$$2 = 1 \cdot 2 + 0 \Longrightarrow (43,35) = 1$$

Bu değerler sondan başa doğru kullanılırsa, x_0 ve y_0 tam sayıları

$$1 = 3 - (8 - 3 \cdot 2) \cdot 1 = 3 \cdot 3 - 8$$

$$1 = (35 - 8 \cdot 4) \cdot 3 - 8 = 3 \cdot 35 - 13 \cdot 8$$
$$1 = 3 \cdot 35 - 13 \cdot (43 - 35.1) = 3 \cdot 35 - 13 \cdot 43 + 13 \cdot 35 = 16 \cdot 35 - 13 \cdot 43$$
$$x_0 = 16, y_0 = -13$$

olarak elde edilir.

2.1.2. Asal sayılar

Tanım 2.3. $p > 1$ bir tam sayı olmak üzere, eğer p'nin 1 ve kendisi dışında pozitif bir böleni yoksa p sayısına asal sayı denir.

Tanım 2.4. p bir asal sayı olmak üzere, eğer $2p + 1$ sayısı da asal ise p sayısına Sophie Germain asalı denir.

Teorem 2.3. Eğer n bileşik sayı ise, n'nin $\sqrt{n}$'yi geçmeyen en az bir asal çarpanı vardır.

Bir sayının asal olup olmadığı, bu teoremden yararlanan basit bölme algoritmasıyla sınanabilir. Algoritmanın çalışma mantığı Örnek 2.3'te gösterilmiştir. Ancak bu yöntem polinom zamanlı değildir; bu nedenle kriptografik uygulamalarda Miller–Rabin gibi polinom zamanlı asal test algoritmaları tercih edilmektedir (Menezes vd., 1997).

Örnek 2.3. 189 sayısının asal sayı olup olmadığı Teorem 2.3. kullanılarak şu şekilde belirlenebilir. $\sqrt{189} < 14$ olduğundan olduğundan, 189 sayısının asal olup olmadığını belirlemek için yalnızca 2,3,5,7,11 ve 13 asal sayılarını denemek yeterlidir. $3|189$ olduğundan, 189 sayısı asal değildir.

2.1.3. Kongrüanslar

Tanım 2.5. a, b ve $m > 0$ tam sayılar olmak üzere, eğer $m|a - b$ ise a, b'ye m modülüne göre kongrüenttir (denktir) denir ve bu duurm

$$a \equiv b \pmod{m}$$

şeklinde yazılır.

Algoritma 2.2. (Soldan sağa modüler üs alma algoritması) Bir sayının kuvvetini hızlı bir şekilde hesaplamak için kullanılan yöntemlerden biri ardışık kare alma yöntemidir.

a^x değeri hesaplanırken öncelikle x sayısı ikilik tabanda $x = (x_n x_{n-1} \dots x_1)_2$ şeklinde yazılır ve daha sonra ardışık kare alma yöntermi ile sırasıyla $a, a^2, a^4, a^8, \dots, a^{2^n}$ değerleri elde edilir. Son aşamada ise $a^x = a^{2^{x_1}} . a^{2^{x_2}} \dots . a^{2^{x_n}}$ eşitliği kullanılarak sonuç elde edilir. Tüm

bu işlemler m modülü altında gerçekleştirilirse algoritma soldan sağa modüler üs alma (Left-to-Right Binary Exponentiation) algoritması olarak adlandırılır. Bu algoritmanın zaman karmaşıklığı

$$O(\log x \log^2 m)$$

şeklindedir.

Teorem 2.4. a, b, c ve $m > 0$ tam sayılar ve $a \equiv b \pmod m$ olmak üzere

i. $a \cdot c \equiv b \cdot c \pmod m$
ii. $a^n \equiv b^n \pmod m$ dir (Altındiş, 2005).

Tanım 2.6. $c_1, c_2, \cdots c_m$ birbirinden farklı tam sayılar olmak üzere eğer her $i \neq j$ için $c_i \not\equiv c_j \pmod m$ oluyorsa $S = \{c_1, c_2, \cdots c_m\}$ kümesine m modülüne göre bir tam kalan sistemi denir.

Örnek 2.4. $\mathbb{Z}_n = \{0,1,2, \dots, n-1\}$ kümesi n modülüne göre bir tam kalan sistemidir.

Tanım 2.7. $S = \{c_1, c_2, \cdots c_m\}$ kümesi m modülüne göre bir tam kalan sistemi olmak üzere $R = \{c_i \in S : (c_i, m) = 1\}$ kümesine m modülüne göre indirgenmiş kalan sistemi denir.

Örnek 2.5. $\mathbb{Z}_n^* = \{a \in \mathbb{Z}_n : (a, n) = 1\}$ kümesi n modülüne göre bir indirgenmiş kalan sistemidir.

Tanım 2.8. (Euler fonksiyonu) m modülüne göre herhangi bir indirgenmiş kalan sisteminin eleman sayısını veren fonksiyona Euler fonksiyonu denir ve $\varphi(m)$ ile gösterilir.

Teorem 2.5. p asal ise $\varphi(p) = p - 1 = \left|\mathbb{Z}_p^*\right|$ dir.

Teorem 2.6. p asal ve a pozitif bir tam sayı ise $\varphi(p^a) = p^{a-1}(p-1)$ dir.

Teorem 2.7. $(m, n) = 1$ pozitif tam sayılar olmak üzere $\varphi(m.n) = \varphi(m) \cdot \varphi(n)$ dir.

Teorem 2.8. $n = {p_1}^{a_1} \cdot {p_2}^{a_2} \cdot \cdots \cdot {p_s}^{a_s}$ ise

$$\varphi(n) = n \cdot \left(1 - \frac{1}{p_1}\right) \cdot \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_s}\right)$$

dir.

Teorem 2.9. (Euler teoremi) m bir pozitif tam sayı ve a bir tam sayı olmak üzere $(a,m) = 1$ olsun. O zaman

$$a^{\varphi(m)} \equiv 1 \pmod{m}$$

dir.

Teorem 2.10. (Fermat teoremi) p bir asal sayı ve a bir pozitif tam sayı olmak üzere $p \nmid a$ olsun. O zaman

$$a^{p-1} \equiv 1 \pmod{p}$$

dir.

2.1.4. Lineer kongrüanslar

Tanım 2.9. x bilinmeyeni göstermek üzere $a.x \equiv b \pmod{n}$ şeklindeki kongrüansa bir bilinmeyenli lineer kongrüans denir.

Teorem 2.11. a ve b tam sayılar ve $(m,a) = d$ olmak üzere $a.x \equiv b \pmod{m}$ kongrüansının çözülebilir olması için gerek ve yeter şart $d|b$ olmasıdır. Eğer bu kongrüans çözülebilirse birbirlerine kongrüent olmayan d tane çözüm vardır.

Sonuç 2.1. $(m,a) = 1$ olsun. Bu takdirde $a.x \equiv 1 \pmod{m}$ kongrüansının tek çözümü vardır.

Örnek 2.6. $35x \equiv 1 \pmod{43}$ kongrüansının çözümünü bulalım.

$$35x - 1 = 43k$$
$$35x - 43k = 1$$

olup bu eşitliğin çözümü Örnek 2.2.'den dolayı $x = 16$ tır.

Tanım 2.10. (Carmichael Lambda fonksiyonu) n pozitif tam sayı olmak üzere, $(a,n) = 1$ olan her a tam sayısı için;

$$a^m \equiv 1 \pmod{n}$$

denkliğini sağlayan en küçük m tam sayısına Carmichael'in Lambda Fonksiyonu denir ve $\lambda(n)$ ile gösterilir (Carmichael 1909, Okumuş 2012).

Teorem 2.12:

- $\lambda(2) = 1, \lambda(4) = 2$ ve $e \geq 3$ için $\lambda(2^e) = 2^{e-2}$
- p tek asal sayı ise $\lambda(p^e) = p^{e-1}(p-1)$
- $n = {p_1}^{e_1}.{p_2}^{e_2} \dots {p_k}^{e_k}$ ise $\lambda(n) = lcm\left(\lambda({p_1}^{e_1})\right), \left(\lambda({p_2}^{e_2})\right), \cdots, \left(\lambda({p_k}^{e_k})\right)$

Teorem 2.13. p ve q asal sayılar olmak üzere;

$$\lambda(p \cdot q) = Lcm(p-1, q-1)$$

dir.

Teorem 2.14. (Çin Kalan Teoremi - CRT) $a_1, a_2, \dots, a_r$ tam sayılar ve $m_1, m_2, \dots, m_r$ ikişer ikişer aralarında asal olan pozitif tam sayılar olmak üzere

$$\begin{aligned} x &\equiv a_1 \pmod{m_1} \\ x &\equiv a_2 \pmod{m_2} \\ &\vdots \\ x &\equiv a_r \pmod{m_r} \end{aligned}$$

lineer kongrüans sisteminin $M = m_1 m_2 \dots m_r$ modülüne göre bir tek çözümü vardır.

Algoritma 2.3. (Gauss's algoritması) Teorem 2.14. ile verilen kongrüans sisteminin çözümü, her $i = 1, 2, \cdots, r$ için $M_i = \frac{M}{m_i}$ ve $M_i . y_i \equiv 1 \pmod{m_i}$ değerleri hesaplanarak

$$x = \left(\sum_{i=1}^{r} M_i \cdot a_i \cdot y_i \right) \pmod{M}$$

şeklinde elde edilir (Menezes vd., 1997).

2.1.5. Lineer olmayan kongrüanslar

Tanım 2.11. a, b ve m tam sayılar, $m \geq 1$ ve $(a, m) = 1$ olsun. O zaman $a^t \equiv 1 \pmod{m}$ kongrüansını sağlayan en küçük pozitif t tam sayısına a'nın m modülüne göre mertebesi denir ve $ord_m a$ ile gösterilir.

Teorem 2.15. $ord_m a = h$ olsun. Bu takdirde $h | \emptyset(m)$ dir.

Tanım 2.12. $ord_m a = \varphi(m)$ ise a'ya m modülüne göre bir ilkel kök denir.

Algoritma 2.4. (İlkel kökün bulunması) p bir asal sayı ve a bir pozitif tam sayı olmak üzere $p \nmid a$ olsun. Teorem 2.10 gereğince

$$a^{p-1} \equiv 1 \pmod{p}$$

olup Teorem 2.15. ve Tanım 2.12. göz önüne alınırsa $d < p$ olan her $d|p-1$ için

$$a^{\frac{p-1}{d}} \not\equiv 1 \pmod{p}$$

sağlanıyorsa a sayısı p modülüne göre ilkel köktür.

Teorem 2.16. a sayısı m modülüne göre bir ilkel kök olsun. Bu durumda

$$a, a^2, \dots, a^{\emptyset(m)}$$

m modülüne göre bir indirgenmiş kalan sistemi oluşturur.

Teorem 2.17. p bir asal sayı, $d \geq 1$ ve $d|p-1$ olsun. O zaman $1 \leq a \leq p-1$ ve $ord_p a = d$ olan a ların sayısı $\emptyset(d)$ dir.

Teorem 2.18. p bir asal sayı olsun. O zaman p modülüne göre $\emptyset(p-1)$ tane ilkel kök vardır.

Tanım 2.13. k pozitif tam sayı ve $(m, a) = 1$ olmak üzere $x^k \equiv a \bmod m$ lineer olmayan kongrüansın çözümü varsa a sayısına m modülüne göre k. kuvvetten kalan(rezidü) denir.

Tanım 2.14. $(a, m) = 1$ olmak üzere $x^2 \equiv a \bmod m$ kongrüansın çözümü varsa a sayısına m modülüne göre kuadratik kalan(rezidü)denir ve $a\mathrm{R}m$ ile gösterilir, çözüm yoksa a sayısına m modülüne göre kuadratik kalan değil (non-rezidü) denir ve $a\mathrm{N}m$ ile gösterilir.

Teorem 2.19. (Euler kriteri) p bir asal tek sayı, $a \in \mathbb{Z}$ ve $(a, p) = 1$ olsun. O zaman

$$x^2 \equiv a \pmod{p}$$

kongrüansının

$$x^{\frac{p-1}{2}} \equiv 1 \pmod{p}$$

ise iki çözümü vardır ve

$$x^{\frac{p-1}{2}} \equiv -1 \pmod{p}$$

ise hiçbir çözümü yoktur.

Teorem 2.20. p ve q farklı asal sayılar, $n = p \cdot q$ ve $y \in \mathbb{Z}_n^*$ olsun.

$$yRn \Leftrightarrow yRp \wedge yRq$$

dir.

Tanım 2.15. (Legendre sembolü) p bir asal tek sayı ve a bir tam sayı olsun.

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & p \nmid a \text{ ve } aRp \\ 0 & p|a \\ -1 & p \nmid a \text{ ve } aNp \end{cases}$$

şeklinde tanımlanan ifadeye Legendre sembolü denir.

Teorem 2.21. p bir asal tek sayı ve a, b tam sayılar olmak üzere $(a,p) = (b,p) = 1$ olsun. O zaman aşağıdaki ifadeler sağlanır.

i. $\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \bmod p$

ii. $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$

iii. $a \equiv b \pmod{p}$ ise $\left(\frac{a}{p}\right) \equiv \left(\frac{b}{p}\right)$

iv. $\left(\frac{a^2}{p}\right) = 1,\ \left(\frac{1}{p}\right) = 1$

v. $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}, \left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$

Teorem 2.22. $p = 4k + 3$ formunda bir asal sayı ve $\left(\frac{a}{p}\right) = 1$ olmak üzere $x^2 \equiv a \bmod p$ kongrüansının çözümü $r \equiv a^{\frac{p+1}{4}} \pmod{p}$ olmak üzere $\mp r$ dir.

İspat:

$$\begin{aligned} r^2 &\equiv a^{\frac{p+1}{2}} \pmod{p} \\ &\equiv a^{\frac{p-1}{2}} \cdot a \pmod{p} \\ &\equiv \left(\frac{a}{p}\right) \cdot a \\ &\equiv a \bmod p \end{aligned}$$

Sonuç 2.2. p ve q $4k+3$ ve formunda asal sayılar ve $n = p.q$ ve $\left(\frac{a}{p}\right) = \left(\frac{a}{q}\right) = 1$ olsun.

$$x^2 \equiv a \pmod{n}$$

kongrüansının dört farklı çözümü

- $m_p \equiv a^{\frac{p+1}{4}} \pmod{p}$
- $m_q \equiv a^{\frac{q+1}{4}} \pmod{q}$
- $y_p \equiv \frac{1}{p} \pmod{q}$
- $y_q \equiv \frac{1}{q} \pmod{p}$

$$x_{1,2,3,4} \equiv \mp\left(y_p \cdot m_q \cdot p \mp y_q \cdot m_p \cdot q\right) \pmod{n}$$

şeklinde elde edilir.

İspat: Sonuç 2.2 nin ispatını şu şekilde elde edebiliriz.

$$m_p^2 \equiv a^{\frac{p+1}{2}} \equiv a^{\frac{p-1}{2}} \cdot a \equiv \left(\frac{a}{p}\right) \cdot a \equiv a \equiv x^2 \pmod{p}$$

olup buradan

$$x = \mp m_p \pmod{p} \tag{2.1}$$

yazılır. Benzer olarak

$$x = \mp m_q \pmod{q} \tag{2.2}$$

olur. (2.1) ve (2.2) lineer kongrüans sistemine CRT uygulanırsa dört farklı çözüm elde edilir.

2.2. Soyut Cebir

Bu bölümdeki temel tanım ve teoremler (Taşçı, 2007) ve (Asar vd., 2009) referanslı kaynaklardan derlenmiştir.

2.2.1. Grup

Tanım 2.16. A boş olmayan bir küme olmak üzere $*: A \times A \to A$ dönüşümüne A üzerinde bir ikili işlem denir ve $(A,*)$ ifadesine A üzerinde bir cebirsel yapı denir.

Tanım 2.17. G boş olmayan bir küme ve bu küme üzerinde bir ikili işlem * olsun. Buna göre eğer aşağıdaki şartlar sağlanırsa $(G,*)$ cebirsel yapısına, bir grup denir.

- Her $a, b \in G$ için $a * b \in G$
- Her $a, b, c \in G$ için $a * (b * c) = (a * b) * c$
- Her $a \in G$ için $a * e = e * a = a$ olacak şekilde bir $e \in G$ vardır.
- e, G' nin birim elemanı olmak üzere her $a \in G$ için $a * a' = a' * a = e$ olacak şekilde $a' \in G$ vardır.

Örnek 2.7. $\mathbb{Z}_n$ kümesi, mod n de tanımlı toplama işlemi ile bir grup oluşturur ve bu grup $(\mathbb{Z}_n, +)$ ile gösterilir.

Örnek 2.8. $\mathbb{Z}_n^*$ kümesi, mod n de tanımlı çarpma işlemi ile bir grup oluşturur ve bu grup $(\mathbb{Z}_n^*, \cdot)$ ile gösterilir.

Tanım 2.18. $(G,*)$ bir grup olmak üzere eğer her $x, y \in G$ için $x * y = y * x$ oluyorsa o zaman G'ye değişmeli (abelyen) grup denir.

Tanım 2.19. $(G,*)$ bir grup olsun. Eğer G kümesi sonlu ise bu gruba sonlu grup denir. Aksi takdirde sonsuz grup olarak adlandırılır. Sonlu bir grubun eleman sayısına bu grubun mertebesi (kardinalite) denir ve $o(G)$ ya da $|G|$ ile gösterilir.

Örnek 2.9. $|\mathbb{Z}_n^*| = \varphi(n)$ dir.

Tanım 2.20. G bir grup, e G'nin birim elemanı ve $a \in G$ olmak üzere $a^t = e$ olacak biçimde bir t pozitif tam sayısı varsa a elemanına sonlu mertebelidir ve bu pozitif t tam sayılarının en küçüğüne a'nın mertebesi denir ve $o(a)$ ya da $|a|$ ile gösterilir. Eğer $o(a) = m$ bir pozitif tam sayı ise $a^m = e$ ve her $0 < k < m$ tam sayısı için $a^k \neq e$ dir.

Tanım 2.21. (Alt grup) $(G,*)$ bir grup ve H, G'nin boş olmayan bir alt kümesi olsun. Eğer H, G grubundaki işleme göre bir grup teşkil ederse H'ya G'nin bir altgrubu denir ve $H \leq G$ şeklinde gösterilir.

Teorem 2.23. (Lagrange teoremi) G bir sonlu bir grup ve H, G'nin bir alt grubu olsun. Bu takdirde $|H||G|$.

Sonuç 2.3. G bir sonlu bir grup ve $a \in G$ olsun. O zaman $o(a)||G|$ ve böylece $a^{|G|} = e$ dir.

2.2.2. Devirli grup

Teorem 2.24. G bir grup ve $a \in G$ olsun. O zaman $\langle a\rangle = \{a^n : n \in \mathbb{Z}\}$ kümesi G'nin bir alt grubudur.

Not: G bir toplamsal grup ise $\langle a\rangle = \{na : n \in \mathbb{Z}\}$ şeklindedir.

Tanım2.22. G bir grup ve $a \in G$ olsun. G'nin $\langle a\rangle = \{a^n : n \in \mathbb{Z}\}$ alt grubuna G'nin a tarafından üretilen devirli alt grubu denir ve $\langle a\rangle$ şeklinde gösterilir.

Tanım 2.23. G bir grup olsun. Eğer $G = \langle a\rangle$ olacak şekilde bir $a \in G$ varsa G'ye a tarafından üretilen bir devirli grup denir ve a'ya G'nin üreteci denir.

Örnek 2.10. Her $n \geq 1$ için $(\mathbb{Z}_n, +)$ grubu devirlidir.

$$\langle \bar{1}\rangle = \{z\bar{1} : z \in \mathbb{Z}\} = \{\bar{z} : z \in \mathbb{Z}\} = \mathbb{Z}_n$$

Sonuç 2.4. Bir k tam sayısının $(\mathbb{Z}_n, +)$ grubunun bir üreteci olması için gerek ve yeter şart $(k, n) = 1$ olmasıdır.

Teorem 2.26. $G = \langle a\rangle$ bir devirli grup ve $o(a) = m$ olsun. Bu takdirde

- $|G| = m$
- $G = \{a, a^2, a^3, \dots, a^{m-1}, a^m = e\}$

dir.

Teorem 2.25. p asal sayı olmak üzere Z_p^* çarpımsal grubu devirlidir.

Algoritma 2.5. (Üretecin bulunması) $\left|Z_p^*\right| = \varphi(p) = p - 1$ olup $g \in Z_p^*$ olmak üzere Sonuç 2.3. gereğince $o(g)|(p-1)$ olacağından g'nin üreteci olup olmadığı şu şekilde belirlenir:

p asal olmak üzere $p-1$ in her asal böleni q_i için $g^{\frac{p-1}{q_i}} \not\equiv 1 \pmod{p}$ oluyorsa $\langle g \rangle = Z_p^*$ dır.

2.2.3. Halka

Tanım 2.24. R boş olmayan bir küme olmak üzere R üzerinde sırasıyla " $+$ " ve " $\cdot$ " ikili işlemleri tanımlı olsun. Eğer aşağıdaki özellikler sağlanıyorsa $(R,+,\cdot)$ cebirsel yapısına bir halka denir.

i. $(R,+)$ bir değişmeli gruptur.
ii. Her $a,b \in R$ için $a \cdot b \in R$ dır.
iii. Her $a,b,c \in R$ için $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ dır.
iv. Her $a,b,c \in R$ için $a \cdot (b+c) = a \cdot b + a \cdot c$ ve $(b+c) \cdot a = b \cdot a + c \cdot a$ dır.

Eğer $(R,+,\cdot)$ cebirsel yapısı bir halka olmak üzere çarpma işlemine göre değişmeli ise yani $\forall\, a,b \in R$ için için $a \cdot b = b \cdot a$ ise halkaya değişmeli halka denir.

Not: Eğer her $a \in R$ için $a \cdot 1_R = 1_R \cdot a = a$ olacak şekilde bir $1_R \in R$ varsa R ye birimli halka denir.

Tanım 2.25. R bir halka olmak üzere her $a \in R$ için $n \cdot a = 0$ olacak şekilde bir n pozitif tam sayısı varsa bu özelliği sağlayan en küçük pozitif n sayısına halkanın karakteristiği denir ve $Kar(R)$ şeklinde gösterilir.

Tanım 2.26. R birimli bir halka $(1_R \neq 0_R)$ ve $0_R \neq a \in R$ olsun. Eğer $a \cdot b = 1_R$ olacak biçimde bir $b \in R$ varsa b'ye a'nın sağ tersi, $c \cdot a = 1_R$ olacak biçimde bir $c \in R$ varsa c'ye a'nın sol tersi denir. Eğer $d \in R$ olmak üzere $a \cdot d = d \cdot a = 1_R$ ise d'ye a'nın tersi ve a'ya da tersinir eleman denir.

Tanım 2.27. R birimli bir halka ve $0_R \neq 1_R$ olsun. Eğer R'nin sıfırdan farklı her elemanı tersinir ise R'ye bir bölme halkası denir.

Tanım 2.28. R bir halka ve $0 \neq a \in R$ olmak üzere $a \cdot b = 0$ olacak şekilde $0 \neq b \in R$ varsa a elemanına sol sıfır eleman denir. Yine $0 \neq a \in R$ olma üzere $b \cdot a = 0$ olacak şekilde $0 \neq b \in R$ varsa a elemanına sağ sıfır bölen denir. Eğer a elemanı hem sağ sıfır bölen hem sol sıfır bölen ise kısaca a'ya sıfır bölen denir.

Tanım 2.29. Birimli, değişmeli ve sıfır bölensiz bir halkaya tamlık bölgesi denir.

Tanım 2.30. Bir R halkası verilsin. R halkası üzerine kurulan tek değişkenli polinomlar kümesi

$$R[x] = \{a_0 + a_1x + a_2x^2 + \cdots + a_nx^n \mid n \in N,\ a_i \in R\}$$

şeklinde tanımlanır. $R[x]$ polinomlar üzerinde bilinen toplama ve çarpma işlemlerine göre bir halka oluşturur.

Tanım 2.31. $p(x) \in F(x)$ olsun. $p(x)$ sabit polinom değil ve her birinin derecesi $p(x)$ in derecesinden küçük iki polinomun çarpımı şeklinde yazılamazsa $p(x)$ e indirgenemez polinom denir.

Tanım 2.32. $p(x) \in F(x)$ olmak üzere $\langle p(x)\rangle = \{p(x).f(x): f(x) \in F(x)\}$ olarak tanımlanır.

Tanım 2.33. $F(x)/\langle p(x)\rangle = \{p(x) + \langle p(x)\rangle : f(x) \in F(x)\}$

2.2.4. Cisim

Tanım 2.34. Birimli ve değişmeli bir halkanın sıfırdan farklı her elemanının çarpma işlemine göre tersi varsa bu halkaya cisim denir ve genel olarak F ile gösterilir.
Eğer F kümesindeki eleman sayısı (yani $|F|$) sonlu ise, bu cisme sonlu cisim veya Galois cismi denir.

Sonuç 2.5. p asal olmak üzere $(\mathbb{Z}_p, +, \cdot)$ halkası bir cisimdir.

Teorem 2.27. $F(x)/\langle p(x)\rangle$ cisimdir $\Leftrightarrow$ $p(x)$ polinomu $F(x)$ üzerinde indirgenemzedir.

Teorem 2.28. F bir sonlu cisim, p bir asal sayı ve $\text{kar}(F) = p$ olsun. Bu takdirde $n \geq 1$ için F, p^n elemanlı bir cisimdir.

2.3. Eliptik Eğriler

Bu bölümdeki bilgiler (Işıklı, 2022) kaynağından özet olarak derlenmiştir.

Tanım: 2.25. F bir cisim, $a_1, a_2, \ldots, a_6 \in K$ olmak üzere

$$E: y^2 + a_1 \cdot x \cdot y + a_3 \cdot y = x^3 + a_2 \cdot x^2 + a_4 \cdot x + a_6 \qquad (2.3)$$

şeklinde tanımlı üçüncü dereceden polinom denklemini sağlayan noktalar kümesine F üzerinde eliptik eğri denir ve (2.3) ifadesi uzun Weierstrass formu olarak bilinir. Bu eşitlik uygun bazı değişken dönüşümü ile

$$E: y^2 = x^3 + a \cdot x + b$$

şeklinde kısa Weierstrass form olarak adlandırılan forma indirgenebilir. Burada $a, b \in F$ katsayıları, cismin elemanlarıdır.

$$\Delta = 4a^3 + 27b^2 \neq 0$$

olduğunda eliptik eğri regüler olup eğrinin çakışan köklere sahip olmadığını ve dolayısıyla geometrik olarak keskin köşe veya kesişen noktalar içermediğini ifade eder. Aritmetik olarak bu noktayı şöyle düşünebiliriz:

- Eğri üzerindeki her noktanın tersinin yansıması vardır.
- Dolayısıyla, $\mathcal{O}$ toplamanın birim elemanıdır.

2.3.1. Eliptik eğrilerde nokta toplama

Temel olarak, bir eliptik eğri üzerindeki iki noktanın toplamı, bu iki noktadan geçen doğrunun eğriyi üçüncü bir noktada kesmesiyle tanımlanır. Bu kesim noktasının x-eksenine göre simetriği, toplamın sonucu olarak kabul edilir. Eğer bir nokta kendisiyle toplanacaksa, bu durumda kullanılan doğru, o noktada eğriye teğet olan doğrudur. Bu doğrunun eğriyi ikinci bir noktada kestiği varsayılır ve yine bu noktanın x-eksenine göre simetriği alınarak toplama işlemi tanımlanır.
Eğer toplama işleminde kullanılan iki noktanın apsisleri eşit, ordinatları zıt işaretliyse (örneğin biri (x, y), diğeri $(x, -y)$, bu durumda bu iki noktadan geçen doğru eğriyi başka bir noktada kesmez. Bu özel durumda, bu iki noktanın toplamı, eliptik eğri üzerindeki toplama işleminin birim elemanı olan sonsuzdaki nokta $\mathcal{O}$ olarak tanımlanır. Bu durum, eliptik eğrinin projektif düzlemdeki geometrik yapısıyla ilgilidir.

Cebirsel olarak, apsisleri eşit olan iki nokta $P = (x, y)$ ve $-P = (x, -y)$ için $P + \mathcal{O} = P$ olur. Diğer durumlar için $P_1(x_1, y_1)$ ve $P_2(x_2, y_2)$ olmak üzere $P_1 + P_2 = (x_3, y_3)$ değerleri aşağıdaki formüller ile hesaplanır.

i. $x_1 \neq x_2$ için:

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$

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

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

ii. $x_1 = x_2$ için:

$$\lambda = \frac{3 \cdot x_1^2 - a}{2y_1}$$

$$x_3 = \lambda^2 - 2 \cdot x_1$$

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

Tanım 2.36. Bir E eliptik eğrisi üzerinde yukarıdaki şekilde tanımlanan toplama işlemi aşağıdaki özellikler ile bir grup yapısı oluşturur. Yani P, Q ve $R \in E$ olmak üzere:

- $P + Q \in E$ dır.
- $(P + Q) + R = P + (Q + R)$ dır.
- $P + \mathcal{O} = P$ dır.
- $P + P' = \mathcal{O}$ olacak şekilde $P' \in E$ vardır.

Tanım 2.37. Eliptik eğri üzerindeki en temel işlem, bir noktanın k skaler sayısı ile çarpılmasıdır. Bu işlem aşağıdaki şekilde tanımlanır:

$$kP = \underbrace{P + P + \cdots + P}_{n\ tane}$$

2.3.2. Sonlu cisimler üzerinde eliptik eğriler

$p > 3$ asal bir sayı, $r \in \mathbb{N}^+$, $q = p^r$ olmak üzere, F_q sonlu bir cisimi için $a, b \in F_q$ olmak üzere

$$y^2 = x^3 + a \cdot x + b \pmod{q}$$

Eğrisi üzerindeki $(x, y) \in F_q \times F_q$ noktalarının oluşturduğu kümeye F_q üzerinde bir eliptik eğri denir ve $E(F_q)$ ile gösterilir.

Algoritma 2.6. $kP \pmod{q}$ değerini hesaplamak için öncelikle sırasıyla

$$P,\ 2P,\ 4P,\ 8P,\ 16P,\ \ldots$$

değerleri elde edilir ve daha sonra Algoritma 2.2'de verilen soldan sağa modüler üs alma yöntemine benzer bir yaklaşım kullanılarak kP noktası hesaplanır. Böylece eliptik eğri üzerinde skaler çarpım işlemi verimli bir şekilde gerçekleştirilmiş olur. Bu yöntemde, mod q altında kP noktasının hesaplanmasının algoritmik zaman karmaşıklığı

$$O(\log k \log^2 q)$$

şeklindedir.

3. YÖNTEM

Bu bölümde dijital imza algoritmalarının temelini oluşturan temel kavramlar ve bu algoritmaların geçerliliği, güvenliği ve özellikleri verilmiştir.

3.1. Temel Kriptografik Kavramlar

Açık mesaj/metin: Şifrelenmemiş verilerdir.

Şifreli mesaj/metin: Şifrelenmiş verilerdir.

Şifreleme: Açık mesajın şifreli mesaja dönüştürülmesidir.

Şifre çözme: Şifreli mesajın açık mesaja dönüştürülmesidir.

Anahtar: Açık mesajların şifrelenmesinde ve/veya şifreli mesajların şifrelerinin çözülmesinde kullanılan sayısal parametrelerdir.

Şifreleme fonksiyonu: Bir açık mesajı şifrelemek için kullanılan matematiksel/algoritmik işlemlerdir. M açık mesaj, ek şifreleme anahtarı ve E şifreleme fonksiyonu olmak üzere şifreleme işlemi $E_{ek}(M)$ olarak ifade edilir.

Şifre çözme fonksiyonu: Şifreli mesajdan açık mesajı elde etmek için kullanılan matematiksel/algoritmik işlemlerdir. C Şifreli mesaj, dk şifre çözme anahtarı ve D şifre çözme fonksiyonu olmak üzere şifre çözme işlemi $D_{dk}(C)$ olarak ifade edilir.

Tek kullanımlık şifreleme: Her bir şifreleme için yalnızca bir kez kullanılacak yeni bir anahtar üretilmesi esasına dayanan şifreleme yöntemidir.

Simetrik şifreleme: Mesajların şifrelenmesi ve şifreli mesajların çözülmesi için aynı anahtarın kullanıldığı şifreleme yöntemidir.

Gizli anahtar: Simetrik şifreleme yöntemlerinde, hem şifreleme hem de şifre çözme işlemlerinde kullanılan ortak anahtara gizli anahtar denir.

Simetrik şifreleme şeması: İlk olarak gizli anahtar belirlenir. M açık mesaj, k gizli anahtar ve E şifreleme fonksiyonu olmak üzere şifreli mesaj $C = E_k(M)$ şeklinde elde edilir ve karşı tarafa gönderilir. Şifreli mesajı alan taraf, D şifre çözme fonksiyonunu kullanarak açık mesajı

$M = D_k(C)$ olarak elde eder. Teorik olarak şifre çözme işlemi, şifreleme işleminin tersidir ve $D_k(E_k(M)) = M$ eşitliği sağlanır. Bu kategoride geliştirilen bazı algoritmalar LUCİFER-IBM (1974), DES-NSA (1977), RC2-RİVEST (1987), AES-NSA (1993), RC6-RİVEST (1998) dir. İşlem süresinin hızlı olması bu kategorideki algoritmaların en önemli avantajlarındandır fakat hem anahtar dağıtım hem de anahtar paylaşım problemlerine sahiptirler.

Anahtar dağıtım problemi: Simetrik şifrelemede kullanılacak gizli anahtarın güvenli şekilde karşı tarafa iletilememesi problemidir. Anahtar ya tarafların önceden bir araya gelmesiyle ya da güvenli bir iletişim kanalı üzerinden belirlenmelidir.

Anahtar paylaşım problemi: Simetrik şifrelemede haberleşmek isteyen tüm kullanıcılar kendi aralarında farklı gizli anahtar belirlemelidir. n tane kullanıcının olduğu bir sistemde $\frac{n(n-1)}{2}$ adet anahtara ihtiyaç vardır. Sistem içerisinde çok sayıda kullanıcı olduğunda anahtar sayısının aşırı artması problemidir.

Asimetrik şifreleme: Mesajları şifrelemek ve şifrelenmiş mesajların şifrelerini çözmek için birbirinden farklı anahtarların kullanıldığı şifreleme yöntemidir. Bu yöntem açık anahtarlı şifreleme (Public-Key Encryption) olarak da bilinir.

Genel/Açık anahtar: Asimetrik şifreleme yöntemlerinde açık mesajın şifrelenmesi için kullanılan anahtardır.

Özel anahtar: Asimetrik şifreleme yöntemlerinde şifreli mesajın şifresini çözmek için kullanılan anahtardır.

Asimetrik şifreleme şeması: İlk olarak şifreli mesajı almak isteyen taraf bir genel–özel anahtar çifti üretir ve genel anahtarı kendisine şifreli mesaj göndermek isteyen taraf ile paylaşır. Şifreli mesajı gönderen taraf, açık mesajı genel anahtarı kullanarak şifreler ve elde edilen şifreli mesajı anahtarı üreten tarafa gönderir. Şifreli mesajı alan taraf ise şifreli mesajın şifresini kendi özel anahtarını kullanarak çözer ve açık mesajı elde eder. Teorik olarak, e genel anahtar, d özel anahtar, E şifreleme fonksiyonu, D şifre çözme fonksiyonu, M açık mesaj ve C şifreli mesaj olmak üzere şifreleme işlemi $C = E_e(M)$ ve şifre çözme işlemi $M = D_d(C)$ şeklinde ifade edilir. Bu durumda $D_d(E_e(M)) = M$ eşitliği sağlanır. Bu kategorideki algoritmalar genel anahtarı herkese açık olarak paylaştığından anahtar dağıtım

problemini, tek bir genel anahtar ile birden fazla şifreli mesaj üretilebildiği için de anahtar paylaşım problemini ortadan kaldırmışlardır.

Deterministik şifreleme: Aynı açık mesaj ve aynı genel anahtar kullanıldığında her zaman aynı şifreli mesajı üreten şifreleme algoritmalarıdır.

Olasılıksal şifreleme: Aynı açık mesaj için aynı genel anahtar kullanılsa bile birbirinden farklı şifreli mesajlar üreten şifreleme algoritmalarıdır.

Semantik güvenlik: Bir şifreki mesajdan açık mesaj hakkında hiçbir anlamlı bilginin çıkarılmamasını garanti eden şifreleme özelliğidir. Olasılıksal şifreleme algoritmaları semantik güvenlik özelliğine sahiptirler.

Kimlik Doğrulama: Kimlik doğrulama (Authentication), bir tarafın gerçekten iddia ettiği kimliğe sahip olduğunu karşı tarafa kanıtlamasını ifade eden temel bir güvenlik bileşenidir. Kimlik doğrulama, bilgisayar sistemlerinde genellikle parola, biyometrik veri, güvenlik token'ları veya dijital sertifikalar aracılığıyla gerçekleştirilir. Amacı, sadece yetkili kişilerin sisteme erişimini sağlamak ve bu kişilerin işlemlerini doğrulamak olup, çoğunlukla şifreleme tabanlı protokollerle desteklenir.

Elektronik imza: Elektronik imza, elektronik ortamda gerçekleştirilen işlemlerde bir kişinin veya kurumun irade beyanını ifade eden hukuki bir kavramdır. Elektronik imza kavramı, kullanılan teknik yöntemden bağımsız olarak, elektronik bir verinin belirli bir tarafa ait olduğunu ve bu verinin onaylandığını göstermeyi amaçlar. Bu bağlamda elektronik imzalar her zaman kriptografik altyapıya dayanmak zorunda değildir; taranmış imza görselleri veya e-posta yoluyla verilen onaylar da elektronik imza kapsamında değerlendirilebilir. Bununla birlikte, kriptografik dijital imza algoritmaları kullanılarak üretilen ve güvenli elektronik imza oluşturma araçları ile desteklenen güvenli elektronik imzalar, Türk hukuk sisteminde 5070 sayılı Elektronik İmza Kanunu uyarınca ıslak imza ile eşdeğer hukuki geçerliliğe sahiptir. Türkiye'de bu tür imzalar, sertifika otoriteleri tarafından sağlanan nitelikli elektronik sertifikalar kullanılarak oluşturulmaktadır.

Dijital imza: Dijital imza, kriptografik yöntemlere dayalı olarak tanımlanan ve elektronik imzanın teknik altyapısını oluşturan bir güvenlik mekanizmasıdır. Temel olarak asimetrik anahtar kriptografisi kullanılarak gerçekleştirilen dijital imza şemaları, bir mesajın

bütünlüğünü, kaynağının doğruluğunu ve inkâr edilemezliğini sağlamayı amaçlar. Dijital imza algoritmaları, imza üretme ve doğrulama işlemlerini matematiksel olarak tanımlar ve kriptografi literatüründe bağımsız bir araştırma alanı olarak ele alınır. Bu sayede, iletinin yetkili bir kaynaktan gönderilip gönderilmediği ve iletim sırasında değiştirilip değiştirilmediği doğrulanabilir.

Dijital İmza Şeması: İlk olarak imzalayan taraf, bir özel–genel anahtar çifti üretir ve genel anahtarı doğrulayıcılar tarafından erişilebilir şekilde kamuya açık olarak paylaşır. İmzalayan taraf, açık mesajın özetini (hash) kendi özel anahtarı ile şifreleyerek dijital imzayı oluşturur. Oluşan imza s, açık mesaj M ile birlikte $\sigma = \{s, M\}$ doğrulayıcı tarafa gönderilir. Doğrulayıcı, imzanın geçerliliğini kontrol etmek için açık mesajın özetini alır ve gönderilen imzayı imzalayanın genel anahtarını kullanarak doğrular.

Deterministik imza: Aynı açık mesaj ve aynı özel anahtar kullanıldığında her zaman aynı imzayı üreten algoritmalarıdır.

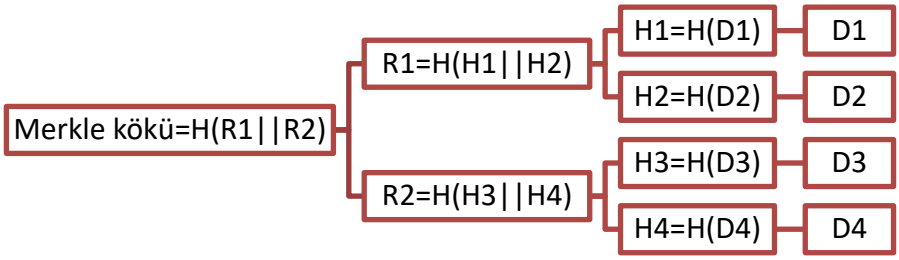
Olasılıksal imza: Aynı açık mesaj ve aynı özel anahtar kullanılsa bile birbirinden farklı şifreli mesajlar üretebilen şifreleme algoritmalarıdır. Bu yönleri ile semantik güvenlik özelliğine sahiptirler.

Kör imza: Kör imza, imzalayan tarafın imzalanan mesajın içeriğini öğrenmeden dijital imza üretmesine olanak tanıyan bir imza mekanizmasıdır. Bu yapı, imza sürecinde gizlilik ve anonimlik gereksinimlerinin ön planda olduğu uygulamalarda kullanılmaktadır. Kör imza kavramı ilk olarak David Chaum tarafından önerilmiş olup, özellikle anonimlik ve gizliliğin kritik olduğu elektronik oylama ve elektronik para sistemleri gibi uygulamalarda önemli bir yere sahiptir.

Kör imza şemaları: Kör imza protokollerinde, imzalanacak mesaj imzalatıcı tarafından bir körleme fonksiyonu kullanılarak dönüştürülür ve bu körlenmiş mesaj imzalayan tarafa gönderilir. İmzalayan, mesajın içeriğini bilmeden körlenmiş mesaj üzerine imza üretir. Daha sonra imzalatıcı, uygulanan körleme işlemini kaldırarak geçerli mesaj–imza çiftini elde eder. Bu süreç sonucunda imzalayan taraf, imzaladığı körlenmiş mesaj ile daha sonra ortaya çıkan gerçek mesaj ve imza arasındaki ilişkiyi kuramaz. Bu özellik sayesinde kör imza şemaları, elektronik oylama sistemlerinde oy gizliliğinin ve bütünlüğünün sağlanmasında, dijital nakit sistemlerinde ise kullanıcı kimliğinin korunmasında yaygın olarak kullanılmaktadır.

Tek Kullanımlık İmza: Tek kullanımlık imza sistemleri (OTS), her anahtar çiftinin yalnızca bir tek mesajın imzalanmasında kullanılabildiği dijital imza şemalarıdır. Bu algoritmalarda, imzalama işlemi yalnızca mesajın özet değeri üzerinden gerçekleştirilir. OTS şemaları yapısal olarak basit olmalarının yanı sıra, kuantum hesaplama saldırılarına karşı dayanıklı olmaları nedeniyle kuantum dayanıklı kriptografi kapsamında özel bir öneme sahiptir. Özellikle yazılım güncellemeleri, donanım üretici sertifikaları ve soğuk cüzdanlar gibi yüksek güvenlik gerektiren senaryolarda uygulanmaktadır. Bununla birlikte, büyük boyutlu anahtar ve imza verisi gibi verimlilik açısından bazı dezavantajlara sahiptir; bu nedenle pratikte sınırlı kullanım alanı bulur. Bu kategorideki ilk örneklerinden biri Lamport OTS şemasıdır.

Merkle kökü: Bir Merkle ağacının (hash ağacının) en üst düğümünü (root node) temsil eden özet değerdir. Alt düzeydeki tüm verilerin özetlerinin ikili biçimde birleştirilip yeniden özetlenmesiyle elde edilir. Bu yapı, veri bütünlüğünü doğrulamak ve büyük veri kümelerinde tek bir hash üzerinden güvenli doğrulama yapmak için kullanılır.



Şekil 1. Dört yapraklı Merkle Kökü

Merkle Doğrulama Yolu (Authentication Path): Merkle doğrulama yolu, bir Merkle ağacında belirli bir yaprak (leaf) ile kök (root) düğüm arasındaki doğrulama işlemini mümkün kılan, gerekli komşu hash değerlerinden oluşan dizidir. Bu yol, imzalanacak yaprağın kök değeri ile ilişkilendirilmesini sağlar. Doğrulama sırasında her seviyede yaprak hash'i ile komşu hashler birleştirilir ve kök hash'e ulaşılır. Hesaplanan kök değeri, Merkle ağacının orijinal kök değeri ile eşleşiyorsa, yaprak ve dolayısıyla imza geçerlidir.

Merkle İmza Şeması (Merkle Signature Scheme, MSS): One-Time Signature (OTS) yalnızca bir kez kullanılabilen bir imza şemasıdır. Her yeni mesaj için yeni bir anahtar çifti gerektiğinden, çoklu imzalar için verimsizdir. Bu problemi çözmek için Merkle ağacı kullanılır. MSS, çok sayıda OTS anahtar çiftini bir Merkle ağacında organize eder. Her OTS anahtar çiftinin genel anahtarı bir yaprak (leaf) düğüm olarak yer alır. Bu yaprakların

hash'leri üst üste hashlenerek Merkle kökü (root) elde edilir ve bu kök, tüm OTS anahtarlarını temsil eden tek bir genel anahtar olarak kullanılır.

İnkâr Edilemez İmza: İnkâr edilemez imza şemaları, imzanın doğrulanması için imzalayanın aktif katılımını gerektiren ve üçüncü taraflarca tek başına doğrulama yapılmasına izin vermeyen özel imza mekanizmalarıdır. Bu yapılarda doğrulayıcı, imzanın geçerliliğini yalnızca imzalayanla etkileşime girerek test edebilir; böylece hem yetkisiz doğrulama girişimleri engellenir hem de imzalayanın imzayı sonradan reddetmesi önlenmiş olur. Gizlilik ve erişim kontrolünün önemli olduğu senaryolar için uygun olan bu şemalar genellikle interaktif protokollerle gerçekleştirilir. Örneğin, güvenli alan erişiminde bir banka müşteriye ait imzanın doğruluğunu müşterinin katılımı olmadan üçüncü bir tarafa kanıtlayamaz; benzer şekilde yazılım dağıtımında, alıcı taraf yazılımın orijinalliğini üretici firmanın katılımı olmaksızın doğrulayamaz. Bu özellikler, izinsiz kullanım ve sahtecilik risklerini azaltan kontrollü bir doğrulama yapısı sunar.

Fail-Stop İmza: Fail-stop imza sistemlerinin temel amacı, bir saldırgan tarafından sahte bir imza üretildiğinde, yasal imzalayıcının bu sahteciliği tespit ederek inkâr edilemez biçimde ispatlayabilmesini sağlamaktır. Bu mekanizmalar, özellikle geleneksel dijital imza algoritmalarının kırıldığı durumlarda, kullanıcıyı suçlamalardan korumak için geliştirilmiştir. Fail-stop yapılar, imzalayan tarafa bir tür "kriptografik güvence" sunar: Sahte bir imza üretilmesi durumunda, imzalayan taraf elinde bulundurduğu ek bilgilerle bu sahteciliği gösterebilir. Bu nedenle, yüksek güvenlik seviyesi gerektiren ve itibarın korunmasının önemli olduğu uygulamalarda kullanılmaktadır.

3.2. Bazı Temel Kriptografik Problemler

Tam sayı çarpanlara ayırma problemi: Yaklaşık olarak aynı büyüklükte iki büyük asal sayının çarpımıyla elde edilen yarı asal bir sayının asal çarpanlarının, bilinen klasik algoritmalar kullanılarak pratik sürede bulunmasının güçlüğünü ifade eden problemdir.

Ayrık logaritma problemi: Yeterince büyük bir asal sayı p ve bir taban g verildiğinde, $g^x \equiv y \pmod{p}$ denklemindeki x değerinin bilinen klasik algoritmalar kullanılarak pratik sürede bulunmasının güçlüğünü ifade eden problemdir.

Eliptik eğri ayrık logaritma problemi: Bir sonlu cisim $\mathbb{F}_p$(veya $\mathbb{F}_{2^p}$) üzerinde tanımlı bir eliptik eğri E ve bu eğri üzerindeki bir nokta P verildiğinde, $Q = kP$ eşitliğini sağlayan k değerinin bilinen klasik algoritmalar kullanılarak pratik sürede bulunmasının güçlüğünü ifade eden problemdir.

Birleşik Mod Altında Karekök Hesaplama Problemi: Yeterince büyük bir bileşik sayı n için, $x^2 \equiv a \pmod{n}$ denklemindeki x değerinin, n'nin asal çarpanları bilinmeksizin pratik sürede bulunmasının güçlüğünü ifade eden problemdir.

Birleşik Mod Altında Yüksek Dereceden Kök Hesaplama Problemi: Yeterince büyük bir bileşik sayı n için, $x^k \equiv a \pmod{n}$ denklemindeki x değerinin, n'nin asal çarpanları bilinmeksizin pratik sürede bulunmasının güçlüğünü ifade eden problemdir.

Günümüzde 2048-bit ve üzeri yarı asal sayılar klasik bilgisayarlar için hâlâ güvenli kabul edililir ancak kuantum bilgisayarların gelişmesiyle birlikte, Shor algoritması (Shor, 1994) gibi kuantum temelli algoritmalar sayesinde bu problemlerin çözülebileceği öngörülmektedir. Bu nedenle, kuantum sonrası kriptografi döneminde söz konusu problemlerin çözümünün mümkün hâle geleceği düşünülmektedir.

Ayrıca NIST SP 800-57 Part 1 Rev. 5 belgesinde yer alan farklı kriptografik problemlerin eşdeğer karşılaştırmaları Tablo 2. de görülebilir.

Tablo 2. Kriptografik problemlerin eşdeğerlik tablosu

Tam sayı çarpanlara ayırma problemi	Eliptik eğri ayrık logaritma problemi
1024-bit	160-223-bit
2048-bit	224-255-bit
3072-bit	256-383-bit
7680-bit	384-511-bit
15360-bit	512-bit

3.3. Temel Şifreleme Algoritmaları

Bu bölümde, klasik dijital imza algoritmalarının temelini oluşturan asimetrik şifreleme algoritmaları ele alınmıştır. İlk olarak, asimetrik kriptografinin ortaya çıkmasına öncülük eden Diffie–Hellman anahtar değişim algoritması açıklanmıştır. Ardından sırasıyla RSA, Rabin ve ElGamal asimetrik şifreleme algoritmalarına yer verilmiştir. Her bir algoritmanın çalışma

prensibi, anahtar üretim aşamaları, şifreleme ve çözme süreçleri adım adım gösterilmiş, geçerliliği cebirsel olarak ispatlanmış ve algoritmaların temel özellikleri ayrıntılı biçimde incelenmiştir.

3.3.1. Diffie-Hellman anahtar değişim algoritması

Simetrik şifrelemede var olan anahtar dağıtım problemini ortadan kaldırmak için Whitfield Diffie ve Martin Hellman tarafından 1976 yılında geliştirilmiştir (Diffie & Hellman,1976). Algoritmanın güvenliği Ayrık Logaritma Problemi'nin çözülmesinin zorluğuna dayanır. Amaç, iki tarafın (A ve B) güvenli bir şekilde ortak bir gizli anahtar (k) oluşturmasını sağlamaktır.

Algoritmanın adımları aşağıdaki gibidir:

1. A ve B tarafları, ortak olarak bir asal sayı p ve bir ilkel kök g üzerinde anlaşırlar.
2. A tarafı gizli bir a değeri seçer ve $A \equiv g^a \bmod p$ değerini hesaplayarak B'ye gönderir.
3. B tarafı gizli bir b değeri seçer ve $B \equiv g^b \bmod p$ değerini hesaplayarak A'ya gönderir.
4. Ortak gizli anahtarı A tarafı, $k \equiv B^a \bmod p$; B tarafı ise $k \equiv A^b \bmod p$ olarak hesaplar.

3.3.2. RSA asimetrik şifreleme algoritması

1978 yılında Ron Rivest, Adi Shamir ve Leonard Adleman tarafından geliştirilen RSA algoritması, hem şifreleme hem de dijital imzalama amacıyla kullanılabilen ilk pratik asimetrik kriptografi algoritmasıdır (Rivest vd., 1978).

$\{p, q\}$ gizli parametreler, $\{n, e\}$ genel anahtar, d özel anahtar, $M \in \mathbb{Z}_n^*$ açık mesaj ve C şifreli mesaj olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- p ve q asal sayılarını seçilir ve $n = p.q$ hesaplanır.
- $\varphi(n) = (p-1)(q-1)$ hesaplanır.
- $1 < e < \varphi(n)$ ve $\big(e, \varphi(n)\big) = 1$ olacak şekilde bir e genel anahtarı seçilir.
- $d \equiv e^{-1} \big(\bmod \varphi(n)\big)$ hesaplanır.

Şifreleme:

- $C \equiv M^e \ (\bmod n)$ hesaplanır.

Şifre çözme:

- $C^d \equiv M \pmod{n}$ hesaplanır.

Algoritmanın geçerliliği:

$$C^d \equiv (M^e)^d \equiv M^{e \cdot d} \equiv M^{\varphi(n) \cdot k+1} \equiv \left(M^{\varphi(n)}\right)^k . M \equiv 1^k . M \equiv M \ (mod\, n)$$

Algoritmanın özelliği:

$$M_1 = M_2 \Rightarrow {M_1}^e \equiv {M_2}^e \Rightarrow C_1 \equiv C_2 (\text{mod}\, n)$$

olduğundan şifreleme algoritması deterministiktir. Ayrıca 1994 yılında Mihir Bellare ve Phillip Rogaway tarafından OAEP (Optimal Asymetric Encryption Padding) olarak adlandırılan olasılıksal versiyonu geliştirilmiştir (Bellare & Rogaway, 1995).

Algoritmanın güvenliği: Tam Sayı Çarpanlara Ayırma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan sırasıyla p, q, $\varphi(n)$ ve d'yi elde eder.

Örnek:

Anahtar üretimi:

- $p = 17$ ve $q = 31$ olsun.
- $n = 527$, $\varphi(n) = 16.30 = 480$,
- $e = 13$ olsun, $d \equiv 13^{-1} \equiv 37 \pmod{480}$

Şifreleme:

- $M = 25$ olsun, $C \equiv 25^{13} \equiv 366 \pmod{527}$

Şifre çözme:

- $366^{37} \equiv 25 \pmod{527}$

3.3.3. Rabin asimetrik şifreleme algoritması

1979 yılında Michael O. Rabin tarafından, RSA algoritmasının bir varyasyonu olarak önerilmiştir (Rabin, 1979). Rabin algoritması, RSA'dan farklı olarak üstel değer olarak sabit $e = 2$ seçilmesi esasına dayanır. Bu durumda şifre çözme işlemi, p ve q asal sayılarına göre modüler kareköklerin alınması ve sonuçların Çin Kalan Teoremi (CRT) yardımıyla birleştirilmesiyle gerçekleştirilir.

$\{p, q\}$ özel anahtar, $n = p \cdot q$ genel anahtar, $M \in \mathbb{Z}_n^*$ açık mesaj ve C şifreli mesaj olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- $p \equiv q \equiv 3 \pmod 4$ olacak şekilde p ve q asal sayıları seçilir ve $n = p.q$ hesaplanır.

Şifreleme:

- $C \equiv M^2 \pmod n$ hesaplanır.

Şifre çözme:

- $m_p \equiv C^{\frac{p+1}{4}} \pmod p$ hesaplanır.
- $m_q \equiv C^{\frac{q+1}{4}} \pmod q$ hesaplanır.
- $y_p \equiv p^{-1} \pmod q$ hesaplanır.
- $y_q \equiv q^{-1} \pmod p$ hesaplanır.
- $M' \equiv \mp\left(y_p.m_q.p \mp y_q.m_p.q\right) \pmod n$ hesaplanır.

Elde edilen 4 farklı M' değerinin biri açık mesajdır. Bu belirsizliği kaldırmak için çeşitli varyasyonlar geliştirilmiştir. Bunlardan biri (Elia vd., 2015) tarafından önerilen padding metodur. Bu metodda $c_1 \equiv M \pmod 2$ ve $c_2 = \left(\frac{M}{n}\right)$ değerlei hesaplanarak şifreli mesajla birlikte çözen tarafa gönderilir. Şifre çözme aşamasında bulunan değerlerden bu eşitlikleri sağlayan M değeri açık mesajdır.

Algoritmanın geçerliliği: Teorem 2.22 ve Sonuç 2.2 ile sağlanır.

Algoritmanın özelliği:

$$M_1 = M_2 \Rightarrow {M_1}^2 \equiv {M_2}^2 \Rightarrow C_1 \equiv C_2 \pmod n$$

olduğundan şifreleme algoritması deterministiktir.

Algoritmanın güvenliği: Hem *Tam Sayı Çarpanlara Ayırma Problemi*'nin hem de *Birleşik Mod Altında Karekök Hesaplama Problemi*'nin zorluğuna dayanır. Eğer Tam Sayı Çarpanlara Ayırma Problemi çözülürse bir saldırgan özel anahtarlar olan p ve q'yu elde eder. Ya da Birleşik Mod Altında Karekök Hesaplama Problemi çözülürse saldırgan direkt olarak açık mesajı elde eder.

Örnek:

Anahtar üretimi:

- $p = 19, q = 31$ ve $n = 589$ olsun.

Şifreleme:

- $M = 25$ olsun, $C = 25^2 \equiv 36 \pmod{589}$
- $c_1 = 25 \equiv 1 \pmod 2$
- $c_2 = \left(\frac{25}{589}\right) = -1$

Şifre çözme:

- $m_p \equiv 36^{\frac{19+1}{4}} \pmod{19} \equiv 6$
- $m_q \equiv 36^{\frac{31+1}{4}} \pmod{31} \equiv 25$
- $y_p \equiv 19^{-1} \pmod{31} \equiv 18$
- $y_q \equiv 31^{-1} \pmod{19} \equiv 8$
- $M_1 \equiv (18.25.19 + 8.6.31) \pmod{589} \equiv 25$
- $M_2 = 589 - 25 = 564$
- $M_3 \equiv (18.25.19 - 8.6.31) \pmod{589} \equiv 583$
- $M_4 = 589 - 583 = 6$

Bu dört sonuçtan $25 \equiv 1 \pmod 2 = 1$ ve $\left(\frac{25}{589}\right) = -1$ olduğundan $M_1 = 25$ çözümdür.

3.3.4. El-Gamal asimetrik şifreleme algoritması

1984 yılında Taher ElGamal tarafından geliştirilen asimetrik şifreleme yöntemidir (ElGamal, 1984). ElGamal algoritması, hem şifreleme hem de dijital imza sistemlerinde kullanılan temel bir yapı sunar.

$\{p, g, y\}$ genel anahtar, x özel anahtar, $M \in \mathbb{Z}_p$ açık mesaj ve $\{C_1, C_2\}$ şifreli mesaj olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- p asal sayısı üretilir ve $\mathbb{Z}_p^*$ nin bir g üreteci seçilir.
- $1 < x < p - 1$ olacak şekilde x özel anahtarı seçilir.
- $y \equiv g^x \pmod p$ genel anahtarı hesaplanır.

Şifreleme:

- $1 < k < p - 1$ olacak şekilde rastgele bir k tam sayısı seçilir.
- $C_1 \equiv g^k \pmod p$ hesaplanır.
- $C_2 \equiv M.y^k \pmod p$ hesaplanır.

Şifre çözme:

- $C_2 . {C_1}^{-x} \equiv M \pmod p$ hesaplanır.

Algoritmanın geçerliliği:

$$C_2 . {C_1}^{-x} \equiv M . y^k . (g^k)^{-x} \equiv M . (g^x)^k . g^{-xk} \equiv M . g^{xk} . g^{-xk} \equiv M \pmod p$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Longrightarrow C_{1,1} \neq C_{1,2} \bigwedge C_{2,1} \neq C_{2,2}$$

olduğundan şifreleme algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Probleminin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod p$ denkliğinden özel anahtar x'i elde eder. Ayrıca farklı iki açık mesaj M_1 ve M_2 aynı k parametresi kullanılarak imzalanırsa

$$\frac{C_{1,2}}{C_{2,2}} \equiv \frac{M_1 . y^k}{M_2 . y^k} \equiv \frac{M_1}{M_2} \pmod p$$

olduğundan saldırgan mesajlardan birini biliyorsa diğerini bulabilir.

Örnek:

Anahtar üretimi:

- $p = 23$, $g = 11$ ve $x = 6$ olsun.
- $y \equiv 11^6 \equiv 9 \pmod{23}$

Şifreleme:

- $M = 10$ ve $k = 3$ olsun.
- $C_1 = 11^3 \equiv 20 \pmod{23}$
- $C_2 = 10 . 9^3 \equiv 22 \pmod{23}$

Şifre çözme:

- $22 . 20^{-6} \equiv 10 \pmod{23}$

3.4. Eliptik Eğri Şifreleme

1985 yılında birbirinden bağımsız olarak Neal Koblitz (Koblitz, 1987) ve Victor Miller (Miller, 1986), eliptik eğrilerin kriptografide kullanılmasını önermiştir. Geleneksel *Ayrık Logaritma Problemi*'ne dayalı şifreleme algoritmalarında kullanılan modüler üs alma işlemi $y = g^x \pmod p$ yerine, sonlu cisimler F_p üzerinde skaler nokta çarpımı $y = x \cdot G \pmod p$ ile işlem yaparak daha kısa anahtarlarla yüksek güvenlik seviyesini sağlamayı

amaçlamışlardır. Bu yaklaşım, özellikle kaynak kısıtlı sistemlerde avantaj sağlamış ve günümüzde eliptik eğri kriptografisi (ECC) adı altında yaygınlaşmıştır (Işıklı, 2022).

3.5. Kriptografik Fonksiyonlar

3.5.1. Hash (Özet) fonksiyonu

Hash fonksiyonları, değişken uzunluktaki bir girdiyi sabit uzunlukta bir çıktıya dönüştüren, aynı girdi için her zaman aynı çıktıyı üreten deterministik matematiksel algoritmalardır. Matematiksel olarak şu şekilde tanımlanır:

$$H: \{0,1\}^* \rightarrow \{0,1\}^n$$

Burada:

- $\{0,1\}^*$: Herhangi bir uzunluktaki bit dizilerini (girdiler) ifade eder.
- $\{0,1\}^n$: Sabit uzunlukta n-bitlik çıktı kümesidir.
- $H(m)$: Mesaj m'nin hash değerini (özetini) temsil eder.

Bir kriptografik hash fonksiyonunun güvenli sayılabilmesi için aşağıdaki üç özelliği sağlaması gerekir:

1. Çakışma Direnci: $\nexists (x, y) : x \neq y \; ve \; H(x) = H(y)$ olmalı.
2. İkinci Örnek Direnci: $\forall x \in \{0,1\}^*$, $H(x) = H(x')$ olacak şekilde $x' \neq x$ bulmak zor olmalı.
3. Ön-İmaj Direnci: $H(x) = h$ verildiğinde x değerini bulmak hesaplanabilir şekilde imkansız olmalı.

Zaman içerisinde çeşitli kriptografik hash algoritmaları geliştirilmiştir. Rivest tarafından tasarlanan MD2 (1989), MD4 (1991) ve MD5 (1992) fonksiyonları bu alandaki öncü çalışmalardandır. Bunları takiben, Amerikan Ulusal Güvenlik Ajansı (NSA) tarafından geliştirilen SHA-1 (1993), SHA-2 (2001) ve SHA-512 (2008) algoritmaları, günümüzde yaygın olarak kullanılan güvenli hash standartları arasında yer almaktadır. Bunun yanında, MDC (Modification Detection Code) tabanlı yapılar, RIPEMD-160 (1996) ve daha ileri güvenlik gereksinimlerini karşılamak üzere tasarlanmış MASH serisi gibi fonksiyonlar da literatürde önemli örnekler olarak gösterilmektedir.

Öte yandan, anahtarlı hash fonksiyonları arasında CBC-MAC, MD5-MAC ve HMAC algoritmaları öne çıkar. Bu fonksiyonlar, özellikle güvenli iletişim protokollerinde mesajın bütünlüğünü ve kimlik doğrulamasını sağlamak amacıyla kullanılmaktadır.

3.5.2. Fazlalık fonksiyonu

Bir mesajın sonuna, mesajın doğruluğunu kontrol etmeye yardımcı olacak ek bir bilgi eklenmesi işlemine *Redundancy Function* (fazlalık fonksiyonu) adı verilir (Roosen, 2015).

Örneğin $m = 1234$ için $r(m) \equiv m \pmod{97}$ şeklinde tanımlanan fazlalık fonksiyonu için $r(1234) = 70 \equiv 1234 \pmod{97}$ olup bu değerin mesaja eklenmesiyle oluşturulan genişletilmiş mesaj $R(m) = 123470$ biçimindedir. Bu yöntem sayesinde mesajın alıcı tarafından doğruluğu kontrol edilebilir hale gelir.

4. BULGULAR

Bu bölümde, literatürde yaygın olarak yer alan klasik dijital imza algoritmaları, kör dijital imza algoritmaları, temel kuantum dayanıklı tek kullanımlık dijital imza algoritmaları ile inkâr edilemez ve *fail-stop* imza algoritmaları ele alınmıştır.

4.1. Klasik Dijital İmza Algoritmaları

Bu bölümde klasik dijital imza algoritmaları incelenmiştir. Ele alınan algoritmalar, imza üretme ve doğrulama süreçlerinde kullanılan matematiksel temeller ve güvenlik varsayımları açısından değerlendirilmiştir. RSA, Rabin, El-Gamal, Schnorr, DSA ve Nyberg–Rueppel gibi yaygın olarak kullanılan dijital imza algoritmalarının yanı sıra bu algoritmaların eliptik eğri tabanlı sürümleri de ele alınarak klasik imza şemalarının genel yapısı ortaya konulmuştur.

4.1.1. RSA imza algoritması

1978 yılında önerilen RSA asimetrik şifreleme algoritmasının dijital imza algoritması olarak kullanılması esasına dayanır. (Rivest vd., 1978)

$\{p, q\}$ gizli parametreler, $\{e, n\}$ genel anahtar, d özel anahtar, $m \in \mathbb{Z}_n$ açık mesaj ve $\sigma = \{m, s\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- p ve q asal sayılarını üretilir ve $n = p.q$ hesaplanır.
- $\varphi(n) = (p-1) \cdot (q-1)$ hesaplanır.
- $1 < e < \varphi(n)$ olacak şekilde $e \in Z^*_{\varphi(n)}$ seçilir.
- $d \equiv e^{-1} \left(\mathrm{mod}\ \varphi(n)\right)$ hesaplanır.

İmzalama:

- $h = H(m)$ hesaplanır.
- $s \equiv h^d \ (\mathrm{mod}\ n)$ hesaplanır.

Doğrulama:

- $h = H(m)$ hesaplanır.
- $s^e \equiv h \ (\mathrm{mod}\ n)$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$s^e \equiv (h^d)^e \equiv h^{d \cdot e} \equiv h^{\varphi(n) \cdot k + 1} \equiv \left(h^{\varphi(n)}\right)^k . h \equiv 1^k . h \equiv h \ (\mathrm{mod}\ n)$$

Algoritmanın özelliği:

$$m_1 = m_2 \Rightarrow h_1 = h_2 \Rightarrow {h_1}^d \equiv {h_2}^d \Rightarrow s_1 \equiv s_2 (\mathrm{mod}\, n)$$

olduğundan imza algoritması deterministiktir.

Algoritmanın güvenliği: Tam Sayı Çarpanlara Ayırma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan sırasıyla p, q, $\varphi(n)$ ve son olarak özel anahtar d'yi elde eder.

Örnek:

Anahtar üretimi:

- $p = 17$ ve $q = 31$ olsun.
- $n = 527$, $\varphi(n) = 16.30 = 480$
- $e = 13$ olsun, $d = 13^{-1} \equiv 37 \pmod{480}$

İmzalama:

- $h = 25$ olsun, $s = 25^{37} \equiv 366 \pmod{527}$

Doğrulama:

- $366^{13} \equiv 25 \pmod{527}$

4.1.2. Rabin imza algoritması

Rabin asimetrik şifreleme algoritması, Michael O. Rabin tarafından 1979 yılında önerilmiş olup, şifre çözme aşamasında kullanılan modüler karekök hesaplama yönteminin imza oluşturma sürecinde uygulanması esasına dayanır (Rabin, 1979).

$\{p, q\}$ özel anahtar, n genel anahtar, $m \in \mathbb{Z}_n$ açık mesaj ve $\sigma = \{s, k\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- $p \equiv q \equiv 3 \pmod 4$ olacak şekilde p ve q asal sayılarını üretilir ve $n = p.q$ hesaplanır.

İmzalama:

- $h = H(m)$ hesaplanır.
- $\left(\frac{h \cdot k}{p}\right) = \left(\frac{h \cdot k}{q}\right) = 1$ olacak şekilde rastgele bir k tam sayısı seçilir.
- $s^2 \equiv h.k \pmod n$ denkliğini sağlayan s değerini Sonuç 2.2 ile hesaplanır.

Doğrulama:

- $s^2 \equiv h.k \pmod n$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği: $s^2 \equiv h.k \pmod{n}$ eşitliğini sağlayan s ancak p ve q kullanılarak Sonuç 2.2 ile hesaplanır.

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow h.k_1 \not\equiv h.k_2 \Rightarrow s_1^2 \not\equiv s_1^2 \pmod{n}$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Hem *Tam Sayı Çarpanlara Ayırma Problemi*'nin hem de *Birleşik Mod Altında Karekök Hesaplama Problemi*'nin zorluğuna dayanır. Eğer Tam Sayı Çarpanlara Ayırma Problemi çözülürse bir saldırgan özel anahtarlar olan p ve q'yu elde eder. Ya da Birleşik Mod Altında Karekök Hesaplama Problemi çözülürse bu durumda da saldırgan direkt olarak sahte imzalar üretebilir.

Örnek:

Anahtar üretimi:

- $p = 43, q = 71$ ve $n = 3053$ olsun.

İmzalama:

- $h = 20$ ve $k = 3$ olsun, $s^2 = 20 \cdot 3 = 60$ ve $\left(\frac{60}{43}\right) = \left(\frac{60}{71}\right) = 1$
- $m_p \equiv 60^{\frac{44}{4}} \pmod{p} \equiv 24,\ m_q \equiv 60^{72/4} \pmod{q} \equiv 29$
- $y_p \equiv \frac{1}{43} \pmod{71} \equiv 38,\ y_q \equiv \frac{1}{71} \pmod{43} \equiv 20$
- $s \equiv (38.29.43 + 20.24.71) \pmod{3053} \equiv 2088$

Doğrulama:

- $2088^2 \equiv 60 \pmod{3053}$

4.1.3. El-Gamal imza algoritması

1984 yılında Taher Elgamal tarafından ElGamal asimetrik şifreleme algoritmasının bir uzantısı olarak önerilmiştir (Elgamal, 1984).

$\{p, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, $\sigma = \{s, r\}$ imza olmak üzere algoritmanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- Yeterince büyük bir p asal sayısı üretilir.
- $\langle g \rangle = \mathbb{Z}_p^*$ olacak şekilde bir g tam sayısı seçilir.

- $1 < x < p - 1$ olacak şekilde rastgele bir x sayısı seçilir.
- $y \equiv g^x \pmod p$ hesaplanır.

İmzalama:

- $1 < k < p - 1$ olacak şekilde rastgele bir k tam sayısı seçilir.
- $r \equiv g^k \pmod p$ hesaplanır.
- $h = H(m)$ hesaplanır.
- $s \equiv (h - x.r).k^{-1} \big(\mathrm{mod}\ (p-1)\big)$ hesaplanır.

Doğrulama:

- $h = H(m)$ hesaplanır.
- $y^r.r^s \equiv g^h \pmod p$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$y^r.r^S \equiv (g^x)^r(g^k)^S \equiv g^{xr}.g^{k(h-xr)k^{-1}} \equiv g^{xr}.g^h.g^{-xr} \equiv g^h \pmod p$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow (h - x.r_1).k_1^{-1} \not\equiv (h - x.r_2).k_2^{-1} \Rightarrow s_1 \neq s_2 \big(\mathrm{mod}\ (p-1)\big)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod p$ denkliğinden özel anahtar x'i elde edilir. Ayrıca farklı iki m_1 ve m_2 mesajı aynı k parametresi kullanılarak imzalanırsa saldırgan;

$$\frac{m_1 - m_2}{s_1 - s_2} \equiv k \pmod{p-1}$$

şeklinde k'yı ve imza algoritması yardımı ile özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 23, g = 11$ ve $x = 6$ olsun.
- $y \equiv g^x \equiv 11^6 \equiv 9 \pmod{23}$

İmzalama:

- $h = 25$ ve $k = 3$ olsun, $r = 11^3 \pmod{23} \equiv 20$
- $s = (25 - 6.20).3^{-1} \pmod{22} \equiv 5$

Doğrulama:

- $11^{25} \equiv 9^{20}.20^5 \pmod{23}$

4.1.4. El-Gamal eliptik eğri imza algoritması

ElGamal dijital imza algoritmasında modüler üs alma yerine eliptik eğri üzerinde modüler skaler çarpımı kullanılan versiyondur.

$\{p, G, Y\}$ genel anahtar, x özel anahtar, m açık mesaj, $\sigma = \{m, r, s\}$ imza olmak üzere algoritmanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- E eliptik eğrisi ve yeterince büyük bir p asal sayısı üretilir.
- Rastgele bir $G \in E$ noktası seçilir.
- $1 < x < p - 1$ şartını sağlayan rastgele bir x sayısı belirlenir.
- $Y \equiv x.G \pmod{p}$ hesaplanır.

İmzalama:

- $1 < k < p - 1$ şartını sağlayan rastgele bir k sayısı seçilir.
- $r \equiv (r_x, r_y) \equiv k.G \pmod{p}$ hesaplanır.
- $h = H(m)$ hesaplanır.
- $s \equiv (h - x.r_x).\mathrm{k}^{-1} \pmod{(p-1)}$ hesaplanır.

Doğrulama:

- $h = H(m)$ hesaplanır.
- $(r_x.Y + s.r) \equiv h.G \pmod{p}$ sağlanıyorsa imza geçerlidir.

İmza algoritmasının geçerliliği:

$$r_x.Y + s.r \equiv r_x.x.G + (h - x.r_x).\mathrm{k}^{-1}.k.G \equiv r_x.x.G + h.G - x.r_x.G \equiv h.G \pmod{p}$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow (h - x.r_{1_x}).k_1^{-1} \not\equiv (h - x.r_{2_x}).k_2^{-1} \Rightarrow s_1 \not\equiv s_2 \pmod{(p-1)}$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği:

Eliptik Eğriler üzerinde Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $Y \equiv x.G \pmod{p}$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $E: y^2 = x^3 - x + 3, p = 79,\ G = (34, 52)$ ve $x = 7$ olsun.

- $Y \equiv 7 \cdot (34, 52) \pmod{79} \equiv (13, 33)$

İmzalama:

- $h = 25$ ve $k = 11$ olsun, $r \equiv 11.(34, 52) \pmod{79} \equiv (30, 11)$
- $s \equiv (25 - 7.30).11^{-1} \pmod{79} \equiv 55$

Doğrulama:

- $25.(34, 52) \pmod{79} \equiv 30.(13, 33) + 55.(30, 11) \pmod{79} \equiv (77, 59)$

4.1.5. Fiege-Fiat-Shamir imza algoritması

1986 yılında Amos Fiat ve Adi Shamir tarafından etkileşimli bir sıfır bilgi ispatını (zero-knowledge proof) dijital bir imzaya dönüştürmek için geliştirilen bir kriptografik tekniktir (Fiat&Shamir, 1986).

$\{p, q\}$ gizli parametreler, n ortak mod, $\{w_1, w_2, \cdots, w_k\}$ genel anahtar, $\{s_1, s_2, \cdots, s_k\}$ özel anahtar, m açık mesaj ve $\sigma = \{m, s\}$ imza olmak üzere algoritmanın adımları aşağıdaki verilmiştir.

Anahtar üretimi:

- p ve q asal sayıları üretilir ve $n = p.q$ hesaplanır.
- $s_1,\ s_2, \cdots,\ s_k \in \mathbb{Z}_n^*$ olacak şekilde rastgele $s_1,\ s_2, \cdots,\ s_k$ tam sayıları seçilir.
- Her $1 \leq j \leq k$ için $w_j \equiv s_j^{-2} \pmod{n}$ hesaplanır.

İmzalama:

- $1 < r < n$ olacak şekilde rastgele bir r tam sayısı seçilir.
- $u \equiv r^2 \pmod{n}$ hesaplanır.
- $h = H(m \parallel u) = (h_1, h_2, \ldots, h_k)_2$ hesaplanır.
- $s \equiv \left(r.\prod_{j=1}^{k} s_j^{h_j}\right) \pmod{n}$ hesaplanır.

Doğrulama:

- $v \equiv s^2.\prod_{j=1}^{k} w_j^{h_j} \pmod{n}$ hesaplanır.
- $H(m \parallel v) = h$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$s^2.\prod_{j=1}^{k} w_j^{h_j} \equiv r^2.\prod_{j=1}^{k} s_j^{2h_j}.\prod_{j=1}^{k} w_j^{h_j} \equiv r^2.\prod_{j=1}^{k} \left(s_j^2.w_j\right)^{h_j} \equiv r^2 \equiv u \pmod{n}$$

Bu durumda $v = u$ ise $H(m \parallel v) = H(m \parallel u)$ dir.

Algoritmanın özelliği:

$$r_1 \neq r_2 \Rightarrow u_1 \not\equiv u_2 \Rightarrow h_1 \not\equiv h_2 \Rightarrow s_1 \not\equiv s_2 (\mathrm{mod}\, n)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Hem *Tam Sayı Çarpanlara Ayırma Problemi*'nin hem de *Modüler Karekök Hesaplama Problemi*'nin zorluğuna dayanır. Modüler karekök hesaplama problemi çözülürse bir saldırgan sahte imzalar üretebilir, benzer olarak Tam Sayı Çarpanlara Ayırma Problemi çözülürse de saldırgan Sonuç 2.2 yardımı ile sahte imzalar üretebilir.

Örnek:

Anahtar üretimi:

- $p = 37, q = 43$ ve $n = 1591$ olsun.
- $s_1 = 267, s_2 = 310, s_3 = 11$
- $w_1 = 267^{-2} \pmod{1591} \equiv 1565$
- $w_2 = 310^{-2} \pmod{1591} \equiv 619$
- $w_3 = 11^{-2} \pmod{1591} \equiv 618$

İmzalama:

- $r = 50$ olsun. $u = 50^2 \pmod{1591} \equiv 909$
- $h_1 = 1, h_2 = 1, h_3 = 0$
- $s = 50.267.310 \equiv 309 \pmod{1591}$

Doğrulama:

- $309^2.1565.619 \equiv 909 \pmod{1591}$

4.1.6. Guillou–Quisquater imza algoritması

1985 yılında Louis C. Guillou ve Jean-Jacques Quisquater sıfır bilgiye dayalı bir kimlik doğrulama protokolünü (GQ) önermişler ve 1988 yılında Fiat-Shamir dönüşümüne dayanan GQ etkileşimsiz dijital imzaya dönüşümünü yayınlamışlardır (Guillou&Quisquater, 1985)**.**

$\{n, e, y, H\}$ genel anahtar, $\{p, q, x\}$ özel anahtar, $m \in Z_n$ açık mesaj ve $\sigma = \{m, s, r\}$ imza olmak üzere algoritmanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- p ve q asal sayıları üretilir ve $n = p.q$ hesaplanır.
- $\big(e, \varphi(n)\big) = 1$ ve $1 < e < \varphi(n)$ olacak şekilde rastgele bir e tam sayısı seçilir.

- $H: \{0,1\}^* \rightarrow \mathbb{Z}_{e-1}$ hash fonksiyonu belirlenir.
- $x \in \mathbb{Z}_n$ olacak şekilde rastgele bir x tam sayısı seçilir.
- $y \equiv x^{-e} \pmod{n}$ hesaplanır.

İmzalama:

- $h = H(m)$ hesaplanır.
- Rastgele bir $k \in \mathbb{Z}_n$ sayısı seçilir.
- $r \equiv k^e \pmod{n}$ hesaplanır.
- $s \equiv k.x^h \pmod{n}$ hesaplanır.

Doğrulama:

- $h = H(m)$ hesaplanır.
- $s^e.y^h \equiv r \pmod{n}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$s^e.y^h \equiv (k.x^h)^e.(x^{-e})^h \equiv k^e \equiv r \pmod{n}$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow k_1.x^h \not\equiv k_2.x^h \Rightarrow s_1 \not\equiv s_2 \pmod{n}$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Hem *Tam Sayı Çarpanlara Ayırma Probleminin* hem de Yüksek Mertebeli Kök Hesaplama Probleminin zorluğuna dayanır. Yüksek Mertebeli Kök Hesaplama Problemi çözülürse bir saldırgan özel anahtar x'i elde eder. Ayrıca Tam Sayı Çarpanlara Ayırma Problemi çözülürse $\big(e, \varphi(n)\big) = 1$ olduğundan $y^{-\frac{1}{e} \pmod{(p-1)}} \equiv x \pmod{p}$ şeklinde özel anahtar elde edilebilir.

Örnek:

Anahtar üretimi:

- $p = 103$, $q = 37$ ve $n = 3811$ olsun.
- $\varphi(n) = 3672$
- $e = 31, x = 7$ için $y \equiv 7^{-31} \pmod{3811} \equiv 3006$

İmzalama:

- $h = 50$ olsun, $k = 13$ için $r = 13^{31} \pmod{3811} \equiv 203$
- $s = 13.7^{50} \pmod{3811} \equiv 561$

Doğrulama:

- $h = 50$ için
- $561^{31}.3006^{50} \pmod{3811} \equiv 203$

4.1.7. Schnorr imza algoritması

1989 yılında Claus-Peter Schnorr tarafından dijital imza oluşturma süreçlerinde güvenli ve verimli bir alternatif olarak önerilmiştir (Schnorr, 1989).

$\{p, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik partametere, $\sigma = \{m, s, r\}$ imza olmak üzere algoritmanın adımları aşağıdaki verilmiştir.

Anahtar üretimi:

- Yeterince büyük bir p asal sayısı üretilir.
- $\langle g \rangle = \mathbb{Z}_p^*$ olacak şekilde rastgele bir g seçilir.
- $1 < x < p - 1$ olacak şekilde rastgele bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

İmzalama:

- $1 < k < p$ olacak şekilde rastgele bir k değeri seçilir.
- $r \equiv g^k \pmod{p}$ hesaplanır.
- $h = H(m)$ hesaplanır.
- $s \equiv (k - x.h) \left(\mathrm{mod}\ (p-1)\right)$ hesaplanır.

Doğrulama:

- $g^s.y^h \equiv r \pmod{p}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$g^s.y^h \equiv g^{k-x.h}.g^{x.h} \equiv g^k \equiv r \pmod{p}$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow (k_1 - x.h) \not\equiv (k_2 - x.h) \Rightarrow s_1 \neq s_2 \left(\mathrm{mod}\ (p-1)\right)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod{p}$ denkliğinden özel anahtar x'i elde eder. Ayrıca farklı iki m_1 ve m_2 mesajı aynı k parametresi kullanılarak imzalanırsa saldırgan;

$$\frac{s_1 - s_2}{m_1 + m_2} \equiv k \pmod{p-1}$$

şeklinde k'yı ve imza algoritması yardımı ile özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 79, g = 13$ ve $x = 23$ olsun.
- $y \equiv g^x \equiv 13^{23} \equiv 9 \pmod{79}$

İmzalama:

- $h = 50$ için $k = 12$ olsun, $r = 13^{12} \pmod{79} \equiv 65$
- $s = (12 - 23.50) \pmod{78} = 32$

Doğrulama:

- $13^{32}.9^{50} \pmod{79} \equiv 65$

4.1.8. Schnorr eliptik eğri imza algoritması

Schnorr dijital imza algoritmasında modüler üs alma yerine eliptik eğri üzerinde modüler skaler çarpımı kullanılan versiyondur.

$\{p, G, Y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik partametere, $\sigma = \{m, s, r\}$ imza olmak üzere algortimanın adımları aşağıdaki verilmiştir.

Anahtar üretimi:

- Uygun bir E eliptik eğrisi belirlenir.
- Yeterine büyük bir p asal sayısı üretilir.
- Rastgele bir $G \in E$ noktası seçilir.
- $1 < x < p - 1$ olacak şekilde rastgele bir x tam sayısı seçilir.
- $Y \equiv x.G \pmod{p}$ hesaplanır.

İmzalama:

- $1 < k < p - 1$ olacak şekilde rastgele bir k sayısı seçilir.
- $r \equiv k.G \pmod{p}$ hesaplanır.
- $h = H(M)$ hesaplanır.
- $s \equiv (k - x.h) \pmod{p}$ hesaplanır.

Doğrulama:

- $h = H(M)$ hesaplanır.
- $r \equiv (s.G + h.Y) \pmod{p}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$s.G + h.Y \equiv (k - x.h).G + h.x.G \equiv k.G - x.h.G + h.x.G \equiv k.G \equiv r \pmod{p}$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow (k_1 - x.h) \not\equiv (k_2 - x.h) \Rightarrow s_1 \neq s_2 \left(\mathrm{mod}\ (p-1)\right)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Eliptik Eğriler üzerinde Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $Y \equiv x.G \pmod{p}$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $E: y^2 = x^3 - x + 3, p = 79,\ G = (34, 52)$ ve $x = 23$ olsun.
- $Y \equiv 23.(34, 52) \pmod{79} \equiv (11, 15)$

İmzalama:

- $k = 12$ olsun. $r \equiv 12.(34, 52) \pmod{79} \equiv (46, 62)$
- $h = 50$ olsun.
- $s \equiv (12 - 23.50) \pmod{79} \equiv 47$

Doğrulama:

- $47.(34, 52) + 50.(11, 15) \pmod{79} \equiv (46, 62)$

4.1.9. Dijital imza algoritması (DSA)

1991 yılında Amerika Birleşik Devletleri Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından Dijital İmza Standardı (Digital Signature Standard - DSS) kapsamında tanımlanmıştır (National Institute of Standards and Technology [NIST], 1991). Algoritma, ElGamal dijital imza şemasının bir türevi olarak tasarlanmıştır.

$\{p, q, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik parametre, $\sigma = \{m, s, r\}$ imza olmak üzere algoritmanın adımları aşağıdaki verilmiştir.

Anahtar üretimi:

- Uygun bir q asal sayısı için $q|(p-1)$ olacak şekilde bir p asal sayısı üretilir.
- Uygun bir $h \in \mathbb{Z}_p^*$ için $g \equiv h^{\frac{p-1}{q}} \not\equiv 1 \pmod{p}$ olacak şekilde g hesaplanır.
- $1 < x < q - 1$ olacak şekilde rastgele bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

İmzalama:

- $1 < k < q - 1$ olacak şekilde rastgele bir k tam sayısı seçilir.
- $r \equiv \left(g^k \pmod p\right) \pmod q$ hesaplanır.
- $h = H(M)$ hesaplanır.
- $s \equiv (h + x.r).k^{-1} \pmod q$ hesaplanır.

Doğrulama:

- $u_1 \equiv s^{-1}.h \pmod q$ ve $u_2 \equiv s^{-1}.r \pmod q$ hesaplanır.
- $\left(g^{u_1}.y^{u_2} \pmod p\right) \equiv r \pmod q$ sağlanıyorsa imza geçerlidir.

Algoritmasının geçerliliği:

İlk olarak $q|(p-1)$ için Fermat teoremi gereğince;

$$g \equiv h^{\frac{p-1}{q}} \pmod p \Rightarrow g^q \equiv h^{p-1} \equiv 1 \pmod p$$

olup bu durumda

$$\begin{aligned} a \equiv b \pmod q \quad &\Leftrightarrow \quad g^a \equiv g^b \pmod p \\ &\Leftrightarrow \quad g^a \equiv \left(g^b \pmod p\right) \pmod q \end{aligned}$$

olduğu açıktır.

$$\begin{aligned} s &\equiv (h + x.r).k^{-1} \pmod q \\ sk &\equiv (h + x.r) \pmod q \\ k &\equiv s^{-1}.(h + x.r) \pmod q \\ &\equiv (u_1 + x.u_2) \pmod q \\ g^k &\equiv \left(g^{u_1 + x.u_2} \pmod p\right) \pmod q \\ r &\equiv \left(g^{u_1}.y^{u_2} \pmod p\right) \pmod q \\ &\equiv r \end{aligned}$$

Algoritmanın özelliği: $k_1 \neq k_2 \Rightarrow s_1 \neq s_2$ olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod p$ denkliğinden özel anahtarı x'i elde eder. Ayrıca farklı iki m_1 ve m_2 mesajı aynı k parametresi kullanılarak imzalanırsa saldırgan;

$$\frac{m_1 - m_2}{s_1 - s_2} \equiv k \pmod{q}$$

şeklinde k'yı ve imza algoritması yardımı ile özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $q = 83, p = 499$ ve $h = 17$ olsun.
- $g \equiv 17^{\frac{498}{83}} \pmod{499} \equiv 440$
- $x = 4$ için $y \equiv 440^4 \pmod{499} \equiv 144$

İmzalama:

- $k = 24$ olsun, $r \equiv \left(440^{24} \pmod{499}\right) \pmod{83} \equiv 26$
- $h = 50$ olsun.
- $s \equiv 45.\{50 + 4.26\} \pmod{83} \equiv 41$

Doğrulama:

- $u_1 \equiv 81.50 \pmod{83} \equiv 66$ ve $u_2 \equiv 26.81 \pmod{83} \equiv 31$
- $\left(440^{66}.144^{31} \pmod{499}\right) \equiv 26 \pmod{83}$

4.1.10. Schnorr imza algoritmasının DSA versiyonu

Schnorr algoritması, DSA algoritmasında kullanılan yöntem ile şu şekilde yazılabilir.

$\{p, q, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik parametre, $\sigma = \{m, s, r\}$ imza olmak üzere algoritmanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- $q|(p-1)$olacak şekilde p ve q asal sayıları üretilir.
- Uygun bir $h \in \mathbb{Z}_p^*$ için $g \equiv h^{\frac{p-1}{q}} \not\equiv 1 \pmod{p}$ olacak şekilde g hesaplanır.
- $1 < x < q - 1$ olacak şekilde rastgele bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

İmzalama:

- $1 < k < q - 1$ olacak şekilde rastgele bir k tam sayısı seçilir.

- $r \equiv \left(g^k \pmod p\right) \pmod q$ hesaplanır.
- $h = H(m, r)$ hesaplanır.
- $s \equiv (k + x.h) \pmod q$ hesaplanır.

Doğrulama:

- $h = H(m, r)$ hesaplanır.
- $\left(g^{s \pmod q}.y^{-h \pmod q} \pmod p\right) \equiv r \pmod q$ sağlanıyorsa imza geçerlidir.

algoritmanın güvenliği, geçerliliği ve özelliği DSA ile benzerdir.

Örnek:

Anahtar üretimi:

- $p = 103, q = 17, h = 2$ olsun, $g \equiv 2^{\frac{(103-1)}{17}} \pmod{103} \equiv 64$
- $x = 13$ olsun, $y \equiv 64^{13} \pmod{103} \equiv 76$

İmzalama:

- $k = 9$ olsun, $r \equiv \left(64^9 \pmod{103}\right) \pmod{17} \equiv 8$
- $h = 25$ olsun, $s \equiv 13.25 + 9 \pmod{17} \equiv 11$

Doğrulama:

- $\left(64^{11 \pmod{17}}.76^{-25 \pmod{17}} \pmod{103}\right) \equiv 8 \pmod{17}$

4.1.11. DSA eliptik eğri imza algoritması

$\{p, q, G, Y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik parametre, $\sigma = \{m, R_x, s\}$ imza olmak üzere algortimanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- E eliptik eğrisi ve uygun bir q asal sayısı için $q|(p-1)$ olacak şekilde bir p asal sayısı üretilir.
- Bir $G \in E$ noktası seçilir.
- $1 < x < q - 1$ şartını sağlayan rastgele bir x tam sayısı belirlenir.
- $Y \equiv x.G \pmod p$ hesaplanır.

İmzalama:

- $1 < k < q - 1$ şartını sağlayan rastgele bir x tam sayısı belirlenir.
- $R \equiv \left(k.G \pmod p\right) \pmod q = \left(R_x, R_y\right)$ hesaplanır. $(R_x \neq 0)$

- $h = H(m)$ hesaplanır.
- $s \equiv k^{-1}.(h + x.R_x) (\text{mod } q)$ hesaplanır.

Doğrulama:

- $h = H(m)$ hesaplanır.
- $u_1 \equiv h/s \pmod q$ ve $u_2 \equiv R_x/s \pmod q$ hesaplanır.
- $\big((u_1.G + u_2.Y) \pmod p\big) \equiv \big(V_x, V_y\big) \pmod q$ hesaplanır.
- $V_x = R_x$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$\begin{aligned} sk \pmod p &\equiv (h + x.R_x) \pmod q \\ k \pmod p &\equiv s^{-1}(h + x.R_x) \pmod q \\ &\equiv (u_1 + x.u_2) (\text{mod } q) \\ k.G &\equiv \big(u_1.G + x.u_2.G \pmod p\big) \pmod q \\ &\equiv \big(u_1.G + u_2.Y \pmod p\big) \pmod q \\ R &\equiv V \end{aligned}$$

Algoritmanın özelliği: $k_1 \neq k_2 \Rightarrow s_1 \neq s_2$ olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Eliptik Eğriler Üzerinde Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $Y \equiv x.G \pmod p$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $E: y^2 = x^3 - x + 3, p = 79, q = 13$ $G = (34, 52)$ ve $x = 7$ olsun.
- $Y \equiv 7 \cdot (34, 52) \equiv (13, 33) \pmod{79}$

İmzalama:

- $k = 11$ ve $h = 25$ olsun.
- $R \equiv 11 \cdot (34, 52) \equiv (30, 11) \pmod{79} = \big(R_x, R_y\big)$
- $s \equiv 11^{-1} \cdot (25 + 7 \cdot 30) \equiv 6 \pmod{13}$

Doğrulama:

- $u_1 \equiv 25/6 \equiv 2 \pmod{13}$ ve $u_2 \equiv 30/6 \equiv 5 \pmod{13}$

- $(V_x, V_y) \equiv (2 \cdot (34,52) + 5 \cdot (13,33)) \equiv (30,11) \pmod{79}$

4.1.12. Nyberg-Rueppel imza algoritması

1993 yılında Kaisa Nyberg ve Rainer A. Rueppel tarafından geliştirilmiştir (Nyberg & Rueppel, 1993). Bu algoritma DSA'ya alternatif olarak daha kısa imzalar üretebilen ve mesaj kurtarma özelliğine sahip olan algoritmalardır.

$\{p, q, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik parametre, $\sigma = \{m, e, s\}$ imza olmak üzere algoritmanın adımları aşağıdaki verilmiştir.

Anahtar üretimi:

- $q|(p-1)$ olacak şekilde yeterince büyük p ve q asal sayıları üretilir.
- Uygun bir $h \in \mathbb{Z}_p^*$ için $g \equiv h^{\frac{p-1}{q}} \not\equiv 1 \pmod{p}$ olacak şekilde g hesaplanır.
- $1 < x < q-1$ olacak şekilde bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

İmzalama:

- Rastgele bir $k \in \mathbb{Z}_P^*$ seçilir.
- $r \equiv g^k \pmod{p}$ hesaplanır.
- $e \equiv R(m).r \pmod{p}$ hesaplanır.
- $s \equiv (x.e + k) \pmod{q}$ hesaplanır.

Doğrulama:

- $g^s.y^{-e} \equiv r \pmod{p}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$g^q \equiv \left(h^{\frac{p-1}{q}}\right)^q \equiv h^{p-1} \equiv 1 \ (mod\ p)$$

olduğundan,

$$g^x \equiv g^{x \bmod q} \ (mod\ p)$$

yazılabilir, bu durumda

$$\begin{aligned} g^s.y^{-e} &\equiv g^{(x.e+k)}.(g^x)^{-e} \pmod{p} \\ & \quad g^k \pmod{p} \\ & \quad r \pmod{p} \end{aligned}$$

Algoritmanın özelliği: Schnorr'dan farklı olarak $\frac{e}{g^s.y^{-e}} \equiv R(m) \pmod q$ olduğundan mesaj kurtarma özelliğine sahiptir. Yani imza olarak mesajın kendisi gönderilmez.

Algortimanın güvenliği: Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod p$ denkliğinden özel anahtarı x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 103, q = 17$ olsun. $(p-1)/q = 6$
- $h = 5$ olsun, $g \equiv 5^6 \equiv 72 \pmod{103}$
- $x = 13$ olsun, $y \equiv 72^{13} \equiv 9 \pmod{103}$

İmzalama:

- $R(m) = 50$ olsun, $k = 19$ için $r \equiv 72^{19} \equiv 34 \pmod{103}$
- $e \equiv (50 \cdot 34) \equiv 52 \pmod{103}$
- $s \equiv (13.52 + 19) \equiv 15 \pmod{17}$

Doğrulama:

- $r \equiv 72^{15} \cdot 9^{-52} \equiv 34 \pmod{103}$

4.2. Kör Dijital İmza Algoritmaları

Bu bölümde kör dijital imza algoritmaları incelenmiştir. RSA, Rabin, Schnorr, DSA ve Nyberg–Rueppel tabanlı kör imza algoritmaları ele alınarak bu şemaların temel çalışma prensipleri ve güvenlik özellikleri değerlendirilmiştir.

4.2.1. RSA kör imza Algoritması

1982 yılında David Chaum tarafından önerilmiştir (Chaum, 1982).

$\{p, q\}$ gizli parametreler, $\{e, n\}$ genel anahtar, d özel anahtar, $m \in \mathbb{Z}_n$ açık mesaj, b körleme faktörü, B kör mesaj, s' kör imza ve $\sigma = \{s, m\}$ imza olmak üzere algoritmanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- p ve q asal sayılarını üretilir ve $n = p.q$ hesaplanır.
- $\varphi(n) = (p-1).(q-1)$ hesaplanır.
- $1 < e < \varphi(n)$ olacak şekilde $e \in Z^*_{\varphi(n)}$ seçilir.

- $d \equiv e^{-1} \left(\mathrm{mod}\ \varphi(n)\right)$ hesaplanır.

Kör Mesaj:

- $b \in Z_n^*$ sayısı seçilir ve $B \equiv m.b^e \pmod{n}$ hesaplanır.

Kör İmza:

- $s_{kör} \equiv B^d \pmod{n}$ hesaplanır.

İmza:

- $s \equiv s_{kör}.b^{-1} \pmod{n}$ hesaplanır.

Doğrulama:

- $s^e \equiv m \pmod{n}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$s^e \equiv (s'.b^{-1})^e \equiv (B^d.b^{-1})^e \equiv B^{e.d}.b^{-e} \equiv B.b^{-e} \equiv m.b^e.b^{-e} \equiv m \pmod{n}$$

Algoritmanın özelliği:

$$b_1 \neq b_2 \Rightarrow m.b_1^e \not\equiv m.b_2^e \Rightarrow B_1^d \not\equiv B_2^d \Rightarrow s'_1 \not\equiv s'_2 \Rightarrow s_1 \not\equiv s_2 (\mathrm{mod}\ n)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Tam Sayı Çarpanlara Ayırma Problemi'nin zorluğuna dayanır. Eğer bu poblemi çözülürse bir saldırgan n ve e sayılarına ulaşabildiği için sırasıyla p, q ve $\varphi(n)$ sayılarını hesaplar ve özel anahtar d'yi elde eder. İmzalayan taraf kör mesaj ile imza arasında bir bağ kuramaz fakat farklı iki m_1 ve m_2 mesajı aynı körleştirme faktörü ile körleştirilirse imzalayan;

$$\frac{B_1}{B_2} \equiv \frac{m_1}{m_2} \pmod{n}$$

denkliğini kontrol ederek bu iki mesajı aynı imzalatanla ilişkilendirebilir.

Örnek:

Anahtar üretimi:

- $p = 11$, $q = 13$, $n = 143$ ve $\varphi(n) = 10.12 = 120$
- $e = 7$ olsun, $d = 7^{-1} \pmod{120} = 103$

Kör Mesaj:

- $m = 50$ olsun.
- $b = 4$ için $B \equiv 50.4^7 \pmod{143} \equiv 96$

Kör İmza:

- $s_{kör} \equiv 96^{103} \pmod{143} \equiv 138$

İmza:

- $s \equiv 138.4^{-1} \pmod{143} \equiv 106$

Doğrulama:

- $106^7 \equiv 50 \pmod{143}$.

4.2.2. Rabin kör imza algoritması

Rabin kör imza algoritması, Rabin imza şeması üzerine, David Chaum'un 1983'te geliştirdiği körleme (blinding) tekniğinin uygulanmasıyla geliştirilmiştir (Elia & Schipani, 2013).

$\{p, q\}$ gizli parametreler, $\{e, n\}$ genel anahtar, $m \in \mathbb{Z}_n$ açık mesaj, b körleme faktörü, B kör mesaj, s' kör imza ve $\sigma = \{s, m\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- $p \equiv q \equiv 3 \pmod{4}$ olacak şekilde p ve q asal sayıları üretilir ve $n = p.q$ hesaplanır.

Körleme:

- $b \in Z_n^*$ sayısı seçilir.
- $B \equiv m.b^2 \pmod{n}$ hesaplanır.

Kör imza:

- $\left(\frac{B \cdot k}{p}\right) = \left(\frac{B \cdot k}{q}\right) = 1$ olacak şekilde rastgele bir k tam sayısı seçilir.
- $s_{kör}^2 \equiv B.k \pmod{n}$ denkliğini sağlayan s' değerini Sonuç 2.2 ile hesaplanır.

İmza:

- $s \equiv s_{kör}.b^{-1} \pmod{n}$ hesaplanır.

Doğrulama:

- $s^2 \equiv m.k \pmod{n}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$s^2 \equiv (s_{kör}.b^{-1})^2 \equiv s_{kör}^2.b^{-2} \equiv B.k.b^{-2} \equiv m.b^2.k.b^{-2} \equiv m.k \pmod{n}$$

Algoritmanın özelliği:

$$b_1 \neq b_2 \Rightarrow B_1 \neq B_2$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Standart Rabin Elektronik İmza Algoritması ve RSA kör dijital imza algoritması ile aynıdır.

Örnek:

Anahtar üretimi:

- $p = 43$, $q = 71$ ve $n = 3053$ olsun.

Körleme:

- $m = 20$ olsun.
- $b = 7$ olsun, $B \equiv 20.7^2 \pmod{3053} \equiv 980$

Kör imza:

- $k = 5$ için
- $B.k = 4900$
- $\left(\frac{4900}{43}\right) = \left(\frac{4900}{71}\right) = 1$
- $m_p \equiv 4900^{44/4} \pmod{43} \equiv 16$
- $m_q \equiv 4900^{72/4} \pmod{71} \equiv 1$
- $y_p \equiv \frac{1}{43} \pmod{71} \equiv 38$
- $y_q \equiv \frac{1}{71} \pmod{43} \equiv 20$
- $s' \equiv (38.1.43 + 20.16.71) \pmod{3053} \equiv 2983$

İmza:

- $s \equiv 2983.7^{-1} \pmod{3053} \equiv 3043$

Doğrulama:

- $3043^2 \equiv 20.5 \pmod{3053} \equiv 100$

4.2.3. Schnorr kör imza algoritması

Bu algoritma, klasik Schnorr dijital imza algoritması üzerine David Chaum'un körleme (blinding) fikrinin eklenmesiyle elde edilmiştir (Fuchsbauer, Plouviez, & Seurin, 2020).

$\{p, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, $\sigma = \{r, s\}$ imza olmak üzere algoritmanın adımları aşağıda verilmiştir.

Anahtar üretimi:

- Yeterince büyük bir p asal sayısı üretilir.
- $\langle g \rangle = \mathbb{Z}_p^*$ olacak şekilde bir g tam sayısı seçilir.

- $1 < x < p - 1$ olacak şekilde rastgele bir x sayısı seçilir.
- $y \equiv g^x \pmod p$ hesaplanır.

Taahhüt:

- $1 < k < p - 1$ olacak şekilde rastgele bir k tam sayısı seçilir.
- $r \equiv g^k \pmod p$ hesaplanır.

Körleme:

- $1 < b < p - 1$ sayısı seçilir.
- $B = r.g^b$
- $h = H(m||B)$ hesaplanır.

Kör İmza:

- $s_{kör} \equiv (x.h + k) \pmod{p - 1}$ hesaplanır.

İmza:

- $s \equiv (s_{kör} + b) \pmod{p - 1}$ hesaplanır.

Doğrulama:

- $g^s \equiv y^h.B \pmod p$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$g^s \equiv g^{s_{kör}+b} \equiv g^{x.h+k+b} \equiv g^{x.h}.g^k.g^b \equiv y^h.r.g^b \equiv y^h.B \pmod p$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow x.h + k_1 + b \not\equiv x.h - k_2 + b \Rightarrow s_1 \neq s_2 \left(\mathrm{mod}\,(p-1)\right)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Problemi'nin zorluığuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod p$ denkliğinden özel anahtar x'i elde eder. Ayrıca farklı iki m_1 ve m_2 mesajı aynı k parametresi kullanılarak imzalanırsa saldırgan;

$$\frac{m_1 - m_2}{s_1 - s_2} \equiv k \pmod{p - 1}$$

şeklinde k'yı ve imza algoritması yardımı ile özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 23, g = 11$ ve $x = 6$ olsun.
- $y \equiv 11^6 \pmod{23} \equiv 9$

Taahhüt:

- $k = 13$ olsun, $r \equiv 11^{13} \pmod{23} \equiv 17$

Körleme:

- $b = 7$ olsun, $B = 17.11^{7} = 331281907$
- $h = 68$

Kör İmza:

- $s_{kör} \equiv (6.68 + 13) \pmod{22} \equiv 3$

İmza:

- $s \equiv (3 + 7) \pmod{22} \equiv 10$

Doğrulama:

- $11^{10} \equiv 9^{68}.331281907 \pmod{23} \equiv 2$

4.2.4. DSA kör imza algoritması

Bu algoritma, klasik DSA üzerine David Chaum'un körleme (blinding) fikrinin eklenmesiyle oluşur (Camenisch, Piveteau, & Stadler, 1994).

$\{p, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik parametre, $\sigma = \{m, r, s\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- Yeterince büyük p ve q asal sayıları üretilir.
- $g \in \mathbb{Z}_p^*$ olacak şekilde g seçilir.
- $1 < x < p - 1$ olacak şekilde bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

Taahhüt:

- Rastgele bir $k \in \mathbb{Z}_P^*$ seçilir.
- $r \equiv g^k \pmod{p}$ hesaplanır.

Körleme:

- $a \in \mathbb{Z}_q$ ve $b \in \mathbb{Z}_q^*$ olacak şekilde a ve b tam sayıları seçilir.
- $R \equiv r^a.g^b \pmod{p}$ hesaplanır.
- $m' \equiv a.m.r.R^{-1} \pmod{p - 1}$ hesaplanır.

Kör İmza:

- $s' \equiv (r.x + k.m') \pmod{p - 1}$ hesaplanır.

İmza:

- $s \equiv (s'.R.r^{-1} + b.m) \pmod{p-1}$ hesaplanır.

Doğrulama:

- $(g^s.y^{-R})^{m^{-1} \pmod{p-1}} \equiv R \pmod{p}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$\begin{aligned}
(g^s.y^{-R})^{m^{-1} \pmod{p-1}} &\equiv (g^s.g^{-x.R})^{m^{-1}} \\
&\equiv \left(g^{s'.R.r^{-1}+b.m}.g^{-x.R}\right)^{m^{-1}} \\
&\equiv \left(g^{(r.x+k.m').R.r^{-1}+b.m}.g^{-x.R}\right)^{m^{-1}} \\
&\equiv \left(g^{x.R+k.m'.R.r^{-1}+b.m}.g^{-x.R}\right)^{m^{-1}} \\
&\equiv (g^{k.a.m+b.m})^{m^{-1}} \\
&\equiv g^{k.a}.g^{b} \\
&\equiv r^{a}.g^{b} \\
&\equiv R \pmod{p}
\end{aligned}$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow r.x + k_1.m' \not\equiv r.x - k_2.m' \Rightarrow s_1 \neq s_2 \left(\mathrm{mod}\,(p-1)\right)$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod{p}$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 149,\ q = 17, g = 64$ ve $x = 13$ olsun.
- $y \equiv 64^{13} \pmod{149} \equiv 133$

Taahhüt:

- $k = 10$ olsun, $r \equiv 64^{10} \pmod{149} \equiv 25$

Körleme:

- $a = 9, b = 7$ ve $m = 71$ olsun.
- $R \equiv 25^9.64^7 \pmod{149} \equiv 47$
- $m' \equiv 9.71.25.47^{-1} \pmod{148} \equiv 25$

Kör imza:

- $s' \equiv (25.13 + 10.25) \pmod{148} \equiv 131$

İmza:

- $s \equiv (131.47.25^{-1} + 7.71) \pmod{148} \equiv 98$

Doğrulama:

- $47 \equiv (64^{98}.133^{-47})^{71^{-1} \pmod{148}} \pmod{149}$

4.2.5. Nyberg-Rueppel kör imza algoritması

1993 yılında Kaisa Nyberg ve Ralph Rueppel tarafından geliştirilmiş bir dijital kör imza algoritmasıdır (Camenisch, Piveteau, & Stadler, 1994).
$\{p, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, k tek seferlik parametre, $\sigma = \{R, m, s\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- Yeterince büyük p ve q asal sayıları üretilir.
- $g \in \mathbb{Z}_p^*$ olacak şekilde g seçilir.
- $1 < x < p - 1$ olacak şekilde bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

Taahhüt:

- Rastgele bir $k \in \mathbb{Z}_P^*$ seçilir.
- $r \equiv g^k \pmod{p}$ hesaplanır.

Körleme:

- $a \in \mathbb{Z}_q$ ve $b \in \mathbb{Z}_q^*$ olacak şekilde a ve b tam sayıları seçilir.
- $R \equiv m.g^a.r^b \pmod{p}$ hesaplanır.
- $m' \equiv r.b^{-1} \pmod{p-1}$ hesaplanır.

Kör İmza:

- $s' \equiv (m'.x + k) \pmod{p-1}$ hesaplanır.

İmza:

- $s \equiv (s'.b + a) \pmod{p-1}$ hesaplanır.

Doğrulama:

- $g^{-s}.y^r.R \equiv m \pmod{p}$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$\begin{aligned} g^{-s}.y^r.R &\equiv g^{-(s'.b+a)}.g^{x.r}.m.g^a.g^{k.b} \\ &\equiv g^{-(m'.x+k).b-a}.g^{x.r}.m.g^a.g^{k.b} \\ &\equiv g^{-(r.b^{-1}.x+k).b-a}.g^{x.r}.m.g^a.g^{k.b} \\ &\equiv m \pmod{p} \end{aligned}$$

Algoritmanın özelliği:

$$k_1 \neq k_2 \Rightarrow m'.x + k_1 \not\equiv m'.x - k_2 \Rightarrow s_1 \neq s_2 \pmod{(p-1)}$$

olduğundan imza algoritması olasılıksaldır.

Algoritmanın güvenliği: *Ayrık Logaritma Problemi*'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod{p}$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 103,\ q = 17, g = 64$ ve $x = 13$ olsun.
- $y \equiv 64^{13} \pmod{103} \equiv 76$

Taahhüt:

- $k = 10$ olsun, $r \equiv 64^{10} \pmod{103} \equiv 100$

Körleme:

- $a = 5$, $b = 7$ ve $m = 101$ olsun.
- $R \equiv 101.64^5.100^7 \pmod{103} \equiv 35$
- $m' \equiv 100.7^{-1} \pmod{102} \equiv 58$

Kör İmza:

- $s' \equiv (58.13 + 10) \pmod{102} \equiv 50$

İmza:

- $s \equiv (50.7 + 5) \pmod{102} \equiv 49$

Doğrulama:

- $101 \equiv 64^{-49}.76^{100}.35 \pmod{103}$

4.3. Kuantum Dayanıklı Temel İmza Algoritmaları

Bu bölümde kuantum saldırılara karşı dayanıklı olduğu kabul edilen temel dijital imza algoritmaları incelenmiştir. Ele alınan imza şemaları, klasik açık anahtarlı sistemlerin aksine tek yönlü özet fonksiyonlara dayalı yapılar kullanmakta ve bu nedenle kuantum bilgisayarların oluşturabileceği tehditlere karşı güvenli alternatifler sunmaktadır. Lamport tek kullanımlık imza algoritması, Merkle–Lamport imza algoritması ve Winternitz tek kullanımlık imza algoritması ele alınarak bu şemaların temel çalışma prensipleri ve kullanım kısıtları değerlendirilmiştir.

4.3.1. Lamport tek kullanımlık imza algoritması

1979 yılında Leslie Lamport tarafından tek kullanımlık imza algoritması olarak önerilmiştir (Lamport, 1979).

pk genel anahtar, sk özel anahtar, m açık mesaj H kriptografik özet (hash) fonksiyonu ve n mesajın bit uzunluğu olmak üzere Lamport tek kullanımlık imza algoritmasının adımları ve özellikleri aşağıda verilmiştir:

Anahtar üretimi:

- Rastgele $2n$ tane sayı üretilir.
  $sk = \{(x_{1,0}, x_{1,1}), (x_{2,0}, x_{2,1}), \dots, (x_{n,0}, x_{n,1})\}$
- Özel anahtardaki her bir sayının hash değeri hesaplanır:
  $$pk = \left\{\left(H(x_{1,0}), H(x_{1,1})\right), \left(H(x_{2,0}), H(x_{2,1})\right), \dots, \left(H(x_{n,0}), H(x_{n,1})\right)\right\}$$
  $$= \{(y_{1,0}, y_{1,1}), (y_{2,0}, y_{2,1}), \dots, (y_{n,0}, y_{n,1})\}$$

İmzalama:

- $H(m) = (h_1 h_2 \dots h_n)_2$ olarak yazılır.
- $s = \{x_{1,h_1}, x_{2,h_2}, \dots, x_{n,h_n}\}$ seçilerek imza oluşturulur.

Doğrulama:

- $H(m) = (h_1, h_2, \dots, h_n)_2$ olarak yazılır.
- Her $i = 1, \dots n$ için $y_{i,h_i} = H(x_{i,h_i})$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği: $\forall\, x \in sk$ için $H(x) \in pk$ olduğundan algoritma geçerlidir.

Algoritmanın özelliği: Lamport imza tamamen deterministiktir. Yani aynı mesaj, aynı özel anahtar ve aynı hash fonksiyonu ile her zaman aynı imza üretilir.

Algoritmanın güvenliği: Kriptografik Hash Fonksiyonlarının Tek Yönlü Olması İlkesi'ne dayanır. Kuantum bilgisayarlara karşı dayanıklıdırlar. Her bir mesaj için yeni bir anahtar kullanılmalıdır. Aksi takdirde $m_1 \neq m_2$ mesajlarının özetleri $H(m_1)$ ve $H(m_2)$ nin en az bir bit pozisyonu farklı olacaktır. İki imza sonunda $k.$ pozisyondaki $x_{k,0}$ ve $x_{k,1}$ değerleri açığa çıkar. Saldırgan yeterince sayıda farklı mesajı aynı anahtar çiftiyle imzalatarak özel anahtarın tüm değerlerini ele geçirebilir.

Örnek: Kolay anlaşılması açısından n =8 bit için CRC-8 hash algoritması tercih edilmiştir.

Anahtar üretimi:

- $sk = \{(100{,}213), (23{,}40), (56{,}82)\}$ olsun.
- $pk = \{(3B, 25), (65, D8), (A8, B9)\}$

İmzalama:

- $H(m) = (101)_2$ olsun.
- $s = \{213{,}23{,}82\}$

Doğrulama:

- $H(m) = (101)_2$
- $H(213) = 25, H(23) = 65, H(82) = B9$ olduğundan imza geçerlidir.

4.3.1.1. Merkle-Lamport imza algoritması

1979 yılında Ralph C. Merkle tarafından tek kullanımlık imza algoritması olarak önerilmiştir (Merkle, 1979).

h Lamport OTS sayısı, $d_{h,1}$ genel anahtar, $sk^{(j)}$ özel anahtrar ve m açık mesajı $n-$bit uzunluğunda olmak üzere

Anahtar Üretimi:

- $t = 2^h$ tane Lamport OTS anahtar çifti üretilir.

$$\left\{sk^{(j)} = \left(x_{i,0}^{(j)}, x_{i,1}^{(j)}\right), \dots, \left(x_{n,0}^{(j)}, x_{n,1}^{(j)}\right)\right\}_{j=1}^{t}$$

$$\left\{pk^{(j)} = \left(y_{i,0}^{(j)}, y_{i,1}^{(j)}\right) = \left(H\left(x_{i,0}^{(j)}\right), H\left(x_{i,1}^{(j)}\right)\right) \mid i = 1, \dots, n\right\}_{j=1}^{t}$$

- Merkle ağacı oluştur:

$$\left\{y^{(j)} = H\left(y_{1,0}^{(j)} \parallel y_{1,1}^{(j)} \parallel y_{2,0}^{(j)} \parallel y_{2,1}^{(j)} \parallel \cdots \parallel y_{n,0}^{(j)} \parallel y_{n,1}^{(j)}\right)\right\}_{j=1}^{t}$$

$$\left\{d_{1,j_1} = H\left(y^{(2j_1-1)} \parallel y^{(2j_1)}\right)\right\}_{j_1=1}^{t/2}$$

$$\left\{d_{h,j_k} = H\left(d_{h-1,2j_{k-1}-1} \parallel d_{h-1,2j_{k-1}}\right)\right\}_{j_k=1}^{t/2^k}, k = 2, \dots, h$$

İmzalama:

- Henüz kullanılmamış bir Lamport özel anahtar $sk^{(j)}$seçilir.

- Lamport OTS kullanılarak mesaj imzalanır: s_j.
- Merkle yolu $p_j = [n_1, n_2, \dots, n_h]$ hesaplanır.
- $\sigma = (s_j, pk^{(j)}, p_j)$

Doğrulama:

- Lamport imzası s_j kullanılarak doğrulanır.
- $y^{(j)} = H\left(y_{1,0}^{(j)} \parallel y_{1,1}^{(j)} \parallel y_{2,0}^{(j)} \parallel y_{2,1}^{(j)} \parallel \cdots \parallel y_{n,0}^{(j)}, \parallel y_{n,1}^{(j)}\right)$ hesaplanır.
- $y^{(j)}$ ve p_j kullanarak merkle kökü r hesaplanır:
- Eğer $r = d_{h,1}$ ise imza geçerlidir.

Algoritmanın özelliği: Aynı Merkle kökü altında t adet imza yapılabilir ve her Lamport imza yalnızca bir kez kullanılabilir.

Algoritmanın güvenliği: Güvenlik, hash fonksiyonunun tek yönlülüğü ve çalışma direnci üzerine kuruludur.

Örnek: Sayısal örnek çok uzun olacağından $n = 3$ ve $h = 2$ için algoritmanın işleyiş yapısı aşağıda verilmiştir.

Anahtar üretimi:

$$sk^{(1)} = \left\{\left(x_{1,0}^{(1)}, x_{1,1}^{(1)}\right), \left(x_{2,0}^{(1)}, x_{2,1}^{(1)}\right), \left(x_{3,0}^{(1)}, x_{3,1}^{(1)}\right)\right\}, \dots, sk^{(4)} = \left\{\left(x_{1,0}^{(4)}, x_{1,1}^{(4)}\right), \dots, \left(x_{3,0}^{(4)}, x_{3,1}^{(4)}\right)\right\}$$

$$pk^{(1)} = \left\{\left(y_{1,0}^{(1)}, y_{1,1}^{(1)}\right), \left(y_{2,0}^{(1)}, y_{2,1}^{(1)}\right), \left(y_{3,0}^{(1)}, y_{3,1}^{(1)}\right)\right\}, \dots, pk^{(4)} = \left\{\left(y_{1,0}^{(4)}, y_{1,1}^{(4)}\right), \dots, \left(y_{3,0}^{(4)}, y_{3,1}^{(4)}\right)\right\}$$

Merkle kökünün hesaplanması:

$$y^{(1)} = \left\{H\left(y_{1,0}^{(1)} \parallel y_{1,1}^{(1)} \parallel y_{2,0}^{(1)} \parallel y_{2,1}^{(1)} \parallel y_{3,0}^{(1)} \parallel y_{3,1}^{(1)}\right)\right\}, \dots, y^{(4)} = \left\{H\left(y_{1,0}^{(4)} \parallel \cdots \parallel y_{3,1}^{(4)}\right)\right\}$$

$$d_{1,1} = \mathrm{H}\left(y^{(1)} \parallel y^{(2)}\right)$$

$$d_{1,2} = \mathrm{H}\left(y^{(3)} \parallel y^{(4)}\right)$$

$$d_{2,1} = \mathrm{H}\left(d_{1,1} \parallel d_{1,2}\right)$$

İmzalama:

$j = 3$ ve $H(m) = (101)_2$ olsun.

$s_3 = \left\{x_{1,1}^{(3)}, x_{2,0}^{(3)}, x_{3,1}^{(3)}\right\}$

$p_3 = \left[d_{1,1}, y^{(3)}\right]$

$\sigma = \left\{s_3, pk^{(3)}, p_3\right\}$

Doğrulama:

$H(m) = (101)_2$ için

$H\left(x_{1,1}^{(3)}\right) = y_{1,1}^{(3)}$

$H\left(x_{2,0}^{(3)}\right) = y_{2,0}^{(3)}$

$H\left(x_{3,1}^{(3)}\right) = y_{3,1}^{(3)}$

$y^{(3)} = H\left(y_{1,0}^{(3)} \parallel y_{1,1}^{(3)} \parallel y_{2,0}^{(3)} \parallel y_{2,1}^{(3)} \parallel y_{3,0}^{(3)}, \parallel y_{3,1}^{(3)}\right)$

$d_{1,2} = \mathrm{H}\left(y^{(3)} \parallel y^{(4)}\right)$

$d_{2,1} = \mathrm{H}\left(d_{1,1} \parallel d_{1,2}\right)$ olduğundan imza geçerlidir.

4.3.2. Winternitz tek kullanımlık imza algoritması

Winternitz tek kullanımlık imza algoritması, 1982 yılında Herbert Winternitz tarafından önerilmiş (Winternitz, 1982) ve tek kullanımlık dijital imza şemalarının temelini oluşturmuştur. Bu algoritma, Merkle ağaçları ile birleştirilerek çoklu imza atılabilen yapılar oluşturmak amacıyla kullanılmış ve literatürde Merkle–Winternitz imza şemaları olarak adlandırılmıştır. Ayrıca, Winternitz algoritması hash tabanlı dijital imza şemalarının çekirdek bileşeni olarak kabul edilmekte olup, XMSS, LMS ve SPHINCS/SPHINCS+ gibi modern kuantum dayanıklı imza algoritmalarının altyapısını sağlamaktadır.

pk genel anahtar, sk özel anahtar, m n-bit uzunluğunda açık mesaj, $v|n$ blok uzunluğu, $l = \frac{n}{v}$ blok sayısı, $w = 2^v$ zincir uzunluğu ve H kriptografik özet (hash) fonksiyonu olmak üzere Winternitz tek kullanımlık imza algoritmasının adımları aşağıda verilmiştir.

Anahtar Üretimi:

- Her $i = 1, \dots, l$ için rastgele x_i seçilir:
- $sk = \{x_i\}_{i=1}^{l}$
- $pk = \left\{y_i = H^{(2^w-1)}(x_i)\right\}_{i=1}^{l}$

İmzalama:

- $H(m) = (h_1, h_2, \dots, h_l)_w$ hesaplanır.
- $\{s_i\}_{i=1}^{l} = \left\{H^{(h_i)}(x_i)\right\}_{i=1}^{l}$hesaplanır.

Doğrulama:

- $H(m) = (h_1, h_2, \dots, h_l)_w$ hesaplanır.
- Her $i = 1,2, \dots, 2^w - h_i - 1$ için $y_i = H^{(i)}(s_i)$ sağlanıyorsa imza geçerlidir.

Algoritmanın geçerliliği:

$$H^{(2^w-h_i-1)}\left(H^{(h_i)}(x_i)\right) = H^{(2^w-1)}(x_i) = y_i$$

Algoritmanın güvenliği: Winternitz imzasının güvenliği, kullanılan hash fonksiyonunun çarpışma ve pre-image dirençliliğine bağlıdır.

Örnek:

Parametre ve Hash fonksiyonu:

- $n = 8, v = 2, l = \frac{8}{2} = 4, w = 2^2 = 4$
- $H(x) = x^2 + 3.x + 3 \pmod{29}$ hash fonksiyonu kullanılsın.

Anahtar Üretimi:

- Her $i = 1, 2, 3, 4$ için: $x_1 = 2, x_2 = 5, x_3 = 9$ ve $x_4 = 16$ olsun.
- y_i ler aşağıdaki gibidir:

$$y_1 = H^{(15)}(2) = 4$$
$$y_2 = H^{(15)}(5) = 2$$
$$y_3 = H^{(15)}(9) = 8$$
$$y_4 = H^{(15)}(16) = 13$$

- Açık anahtar: $pK = (4, 2, 8, 13)$
- Gizli anahtar: $sK = (2, 5, 9, 16)$

İmzalama:

- m mesajı için $H(m) = (1\ 0\ 1\ 1)$
- s_i ler aşağıdaki gibidir:

$$s_1 = H^{(1)}(2) = 13$$
$$s_2 = H^{(0)}(5) = 5$$
$$s_3 = H^{(1)}(9) = 24$$
$$s_4 = H^{(1)}(16) = 17$$

- $s = (13, 5, 24, 17)$

Doğrulama:

- $H(m) = (1\ 0\ 1\ 1)$
- y_i ler aşağıdaki gibidir.

$$y_1 = H^{(14)}(13) = 4$$
$$y_2 = H^{(15)}(5) = 2$$
$$y_3 = H^{(14)}(24) = 8$$
$$y_4 = H^{(14)}(17) = 13$$

olduğundan imza geçerlidir.

4.4. Chaum-van-Antwerpen inkar edilemez imza algoritması

1989 yılında David Chaum ve T. P. van Antwerpen tarafından geliştirilen inkar edilemez imza algoritmasıdır (Chaum & Antwerpen, 1989).

$\{p, g, y\}$ genel anahtar, x özel anahtar, m açık mesaj, $\sigma = \{m, s\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- Uygun bir q asal sayısı için $p = 2q + 1$ olacak şekilde bir p asal sayısı üretilir.
- $g \equiv h^2 \not\equiv 1 \pmod{p}$ olacak şekilde rastgele bir $h \in \mathbb{Z}_p$ sayısı seçilir.
- $1 < x < q - 1$ olacak şekilde rastgele bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

İmzalama: $m^q \equiv 1 \bmod p$ olacak şekilde bir m mesajı aşağıdaki işlem ile imzalanır:

- $S \equiv m^x \pmod{p}$ hesaplanır.

Doğrulama:

1-Doğrulayan taraf:

- $1 < x_1, x_2 < q - 1$ olacak şekilde rastgele x_1 ve x_2 tam sayılarını seçer.
- $z \equiv S^{x_1} . y^{x_2} \pmod p$ hesaplar.

2-İmzalayan taraf:

- $w \equiv z^{x^{-1} \pmod q} \pmod p$ hesaplar.

3-Doğrulayan taraf:

- $w \equiv m^{x_1} . g^{x_2} \pmod p$ olduğundan imza geçerlidir.

Algoritmanın geçerliliği:

$$
\begin{aligned}
w \pmod p &\equiv z^{x^{-1} \pmod q} \pmod p \\
&\equiv (S^{x_1} . y^{x_2})^{x^{-1} \pmod q} \pmod p \\
&\equiv (m^{x \cdot x_1} . g^{x \cdot x_2})^{x^{-1} \pmod q} \pmod p \\
&\equiv m^{x \cdot x_1 \cdot x^{-1} \pmod q} . g^{x \cdot x_2 \cdot x^{-1} \pmod q} \pmod p \\
&\equiv m^{\left(x \cdot x^{-1} \pmod q\right) \cdot x_1} . g^{\left(x \cdot x^{-1} \pmod q\right) \cdot x_2} \pmod p \\
&\equiv m^{(q.k+1).x_1} . g^{(q.k+1).x_2} \pmod p \\
&\equiv m^{q.k.x_1} . m^{x_1} . h^{2q.k.x_2 + 2.x_2} \pmod p \\
&\equiv (m^q)^{k \cdot x_1} m^{x_1} . (h^{p-1})^{k.x_2} . h^{2x_2} \pmod p \\
&\equiv m^{x_1} . h^{2 \cdot x_2} \pmod p \\
&\equiv m^{x_1} . g^{x_2} \pmod p \\
&\equiv w' \pmod p
\end{aligned}
$$

Algoritmanın özelliği:

$$m_1 = m_2 \Rightarrow {m_1}^x \equiv {m_2}^x \pmod p$$

olduğundan imza algoritması deterministiktir.

Algoritmanın güvenliği: Ayrık Logaritma Problemi'nin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod p$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $q = 29, p = 59$ ve $h = 17$ olsun, $a \equiv 17^2 \pmod{59} \equiv 53$
- $x = 3$ için $y \equiv 53^3 \pmod{59} \equiv 20$

İmzalama:

- $m = 20$ olsun.
- $s \equiv 20^3 \pmod{59} \equiv 35$

Doğrulama:

- $x_1 = 18, x_2 = 15$
- $z \equiv 35^{18} . 20^{15} \pmod{59} \equiv 57$
- $w \equiv z^{\left(\frac{1}{3} \bmod 29\right)} \pmod{59} \equiv 21$
- $20^{18} . 53^{15} \equiv 21 \pmod{59}$

4.5. GMR Fail-Stop imza algoritması

1993 yılında Shafi Goldwasser, Silvio Micali ve Charles Rackoff tarafından Fail-Stop imza algoritması olarak önerilmiştir (Goldwasser vd., 1993).

$\{p, g, y, T_1, T_2\}$ genel anahtar, $\{x_1, x_2, y_1, y_2\}$ özel anahtar, $m \in Z_q$ açık mesaj, $\sigma = \{T_1, T_2, s_1, s_2\}$ imza olmak üzere algoritmanın adımları ve özellikleri aşağıda verilmiştir.

Anahtar üretimi:

- Uygun bir q asal sayısı için $q|(p-1)$ olacak şekilde bir p asal sayısı üretilir.
- Uygun bir $h \in \mathbb{Z}_p^*$ için $g \equiv h^{\frac{p-1}{q}} \not\equiv 1 \pmod{p}$ hesaplanır.
- $1 < x < q - 1$ olacak şekilde ratgele bir x tam sayısı seçilir.
- $y \equiv g^x \pmod{p}$ hesaplanır.

İmzalama:

- Rastgele $1 < x_1, x_2, y_1, y_2 < p$ tam sayıları seçilir.
- $T_1 \equiv g^{x_1} . y^{x_2} \pmod{p}$ ve $T_2 \equiv g^{y_1} . y^{y_2} \pmod{p}$ hesaplanır.
- $s_1 \equiv x_1 + m . y_1 \pmod{p}$ hesaplanır.
- $s_2 \equiv x_2 + m . y_2 \pmod{p}$ hesaplanır.

Doğrulama:

- $v_1 \equiv T_1 . {T_2}^m \pmod{p}$ hesaplanır.
- $v_2 \equiv g^{s_1} . y^{s_2} \pmod{p}$ hesaplanır. $v_1 = v_2$ ise imza geçerlidir

Algoritmanın geçerliliği:

$$v_1 \equiv T_1 . {T_2}^m \equiv (g^{x_1} . y^{x_2})(g^{y_1} . y^{y_2})^m \equiv g^{x_1 + y_1 . m} \cdot y^{x_2 + y_2 . m} \equiv g^{s_1} . y^{s_2} \equiv v_2 \pmod{p}$$

Algoritmanın özelliği: x_1, x_2, y_1, y_2 rastgele seçildiğinden imza algoritması olasılıksaldır.

Algoritmanın güvenliği: Ayrık Logaritma Probleminin zorluğuna dayanır. Eğer bu problem çözülürse bir saldırgan $y \equiv g^x \pmod{p}$ denkliğinden özel anahtar x'i elde eder.

Örnek:

Anahtar üretimi:

- $p = 7687$ ve $q = 61$ olsun, $(p-1)/q = 126$
- $h = 17$ olsun, $g \equiv 17^{126} \pmod{7687} \equiv 1141$
- $x = 4$ olsun, $y \equiv 1141^4 \pmod{7687} \equiv 7223$

İmzalama:

- $m = 50$ olsun.
- $x_1 = 7, x_2 = 10, y_1 = 11, y_2 = 26$ olsun.
- $T_1 \equiv 1141^7 . 7223^{10} \pmod{7687} \equiv 7060$
- $T_2 \equiv 1141^{11} . 7223^{26} \pmod{7687} \equiv 2197$
- $s_1 \equiv 7 + 50.11 \pmod{7687} \equiv 557$
- $s_2 \equiv 10 + 50.26 \pmod{7687} \equiv 29$

Doğrulama:

- $v_1 \equiv 7060 \cdot 2197^{50} \pmod{7687} \equiv 2778$
- $v_2 \equiv 1141^{557} \cdot 7223^{1310} \pmod{7687} \equiv 2778$

5. TARTIŞMA VE SONUÇ

Bu tezde klasik dijital imza algoritmaları; cebirsel yapılarına, güvenlik temellerine ve algoritmik özelliklerine göre sistematik biçimde incelenmiştir. Çalışma kapsamında klasik asimetrik imza sistemlerinden başlayarak, eliptik eğri tabanlı ve kuantum sonrası döneme uyumlu imza algoritmalarına kadar geniş bir yelpazede analiz yapılmıştır. Böylece hem tarihsel hem de teorik olarak dijital imza sistemlerinin gelişim süreci bütüncül bir bakış açısıyla değerlendirilmiştir.

Dijital imza algoritmalarının temelinde, sayı kuramı ve soyut cebir kavramlarının güçlü biçimde yer aldığı görülmüştür. Özellikle bölünebilme, asal sayılar, kongrüanslar ve grup teorisi, hem şifreleme hem de imzalama işlemlerinin matematiksel doğruluğunu sağlayan temel taşlardır. Çalışmada bu kavramlar kullanılarak her bir imza algoritmasının geçerliliği cebirsel olarak ispatlanmıştır.

Klasik dijital imza algoritmaları incelendiğinde, algoritmaların çoğunun olasılıksal yapıda olduğu görülmüştür. Özellikle RSA ve Rabin algoritmaları, aynı mesaj için aynı imzayı üretir. Bu durum, teorik olarak ispatlanabilirlik açısından avantaj sağlasa da, pratik uygulamalarda mesaj tekrarı ve imzanın yeniden kullanımına karşı zafiyet oluşturabilir. Buna karşın ElGamal ve Schnorr gibi algoritmalar, rastgelelik (nonce veya rastgele k) kullanarak olasılıksal bir imzalama süreci sunar. Bu yöntem, aynı mesaj için farklı imzalar üretmeyi sağlar ve kriptografik güvenliği artırır.

Güvenlik açısından değerlendirildiğinde, klasik imza algoritmalarının tamamı belirli matematiksel problemlerin çözümündeki zorluğa dayanır. RSA ve Rabin sistemleri, tam sayı çarpanlara ayırma probleminin zorluğuna; ElGamal, DSA ve Schnorr algoritmaları ise ayrık logaritma probleminin çözümündeki güçlüğe dayanır. Eliptik eğri tabanlı versiyonlar, aynı güvenliği daha küçük anahtar boyutlarıyla sağlayarak hem hız hem de verimlilik açısından avantaj sunar.

Çalışmanın bir kısmı, kör imza algoritmalarının incelenmesine ayrılmıştır. Kör imza mekanizması, imza sahibinin mesajın içeriğini görmeden imzalama yapabilmesini sağlar. Bu özellik, gizlilik gerektiren elektronik oylama, anonim kimlik doğrulama ve dijital para sistemlerinde büyük önem taşımaktadır. Tez kapsamında RSA, DSA, Schnorr ve Nyberg–

Rueppel algoritmalarının kör imza versiyonları incelenmiş; körleştirme ve körlükten çıkarma adımlarının cebirsel olarak geçerliliği ispatlanmıştır.

Kuantum bilgisayarların gelişimiyle birlikte, klasik imza algoritmalarının güvenlik temelleri tehdit altına girmiştir. Bu nedenle tezde Lamport, Merkle–Lamport ve Winternitz gibi tek kullanımlık temel imza algoritmaları da değerlendirilmiştir. Bu algoritmaların, kuantum sonrası döneme uygun olarak hash fonksiyonlarına dayalı oldukları ve klasik problemlerin aksine çarpanlara ayırma veya ayrık logaritma gibi problemlere bağımlı olmadıkları belirlenmiştir. Dolayısıyla bu algoritmalar, kuantum bilgisayarlara karşı dayanıklı bir imza altyapısı sunmaktadır. Ancak bu sistemlerin temel dezavantajı, imzalama sürecinde büyük anahtar ve imza boyutlarının ortaya çıkmasıdır.

Çalışmanın genel sonucu olarak, dijital imza algoritmalarının güvenliği yalnızca cebirsel bir doğrulukla değil, aynı zamanda olasılıksal yapı, karmaşık fonksiyon seçimi ve anahtar yönetimi unsurlarıyla birlikte değerlendirilmelidir. Klasik sistemler güçlü matematiksel temellere dayansa da, kuantum çağında güvenliğin sürdürülebilirliği açısından post-kuantum algoritmalara yönelimin kaçınılmaz olduğu görülmüştür.

Sonuç olarak, bu tez çalışması dijital imza algoritmalarının cebirsel yapısını, güvenlik temellerini ve deterministik–olasılıksal karakterlerini matematiksel olarak ortaya koymuştur. Çalışma, hem klasik hem de post-kuantum yaklaşımları karşılaştırmalı biçimde değerlendirerek gelecekteki güvenli dijital imza sistemlerinin teorik altyapısına katkı sağlamayı hedeflemiştir.

KAYNAKÇA

Altındiş, H. (2005) "Sayılar Teorisi ve Uygulamaları", 2. Baskı, Erciyes Üniversitesi Fen Fakültesi Matematik Bölümü, Kayseri, 19-193.

Arıkan, A. & Hacısalihoğlu, H. (2015). Soyut matematik. Hacısalihoğlu Yayınları.

Asar, O., Arıkan,. A., ve Arıkan, A., (2009) "Cebir", Eflatun yayınevi , Ankara.

Asar, A. O. ve Arıkan, A. (2012) Sayılar Teorisi, Gazi kitapevi, Ankara, 9-100.

Bellare, M., & Rogaway, P. (1995). Optimal asymmetric encryption. In Advances in Cryptology—EUROCRYPT'94: Workshop on the Theory and Application of Cryptographic Techniques Perugia, Italy, May 9–12, 1994 Proceedings 13 (pp. 92-111). Springer Berlin Heidelberg.

Boneh, D., & Franklin, M. (2001). Identity-based encryption from the Weil pairing. *Advances in Cryptology—CRYPTO 2001*, 2139, 213–229.

Camenisch, J.-L., Piveteau, J.-M., & Stadler, M. A. (1994). *Blind signatures based on the discrete logarithm problem*. Rump Session of EUROCRYPT '94.

Chaum, D. (1983, August). Blind signatures for untraceable payments. In *Advances in Cryptology: Proceedings of Crypto 82* (pp. 199-203). Boston, MA: Springer US.

Chaum, D., & Van Antwerpen, H. (1989). Undeniable Signatures, in 'Advances in Cryptology–CRYPTO'89', Vol. 435 of LNCS.

Diffie, W. and Hellman, M. (1976) "New directions in crytography", IEEE Transaction on Informations Theory, 644-654.

ElGamal, T. (1985) "A public key cryptosystem and a signature scheme based on discrete logarithms", *IEEE transactions on information theory*, 31(4), 469-472.

Elia, M., & Schipani, D. (2013). *On the Rabin Signature* (Journal of Discrete Mathematical Sciences and Cryptography, 16(6), 367-378).

Elia, M., Piva, M. and Schipani, D. (2015) "The Rabin cryptosystem revisited", Applicable Algebra in Engineering, Communication and Computing, 26, 251 275.

Fiat, A., & Shamir, A. (1986, August). How to prove yourself: Practical solutions to identification and signature problems. In *Conference on the theory and application of cryptographic techniques* (pp. 186-194). Berlin, Heidelberg: Springer Berlin Heidelberg.

Fuchsbauer, G., Plouviez, A., & Seurin, Y. (2020). *Blind Schnorr Signatures and Signed ElGamal Encryption in the Algebraic Group Model*. In Advances in Cryptology –

EUROCRYPT 2020 (Vol. 12106, pp. 63-95). Springer. https://doi.org/10.1007/978-3-030-45724-2_3 (repositum.tuwien.at)

Geneş, N. (2024). Tam sayı çarpanlara ayırma problemine dayalı asimetrik şifreleme algoritmaları (Yüksek lisans tezi, Erzincan Binali Yıldırım Üniversitesi).Yükseköğretim kurulu Ulusal Tez Merkezi.

Goldwasser, S., Micali, S., & Rivest, R. L. (1988). A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on computing*, *17*(2), 281-308.

Guillou, L. C., & Quisquater, J. J. (1990). A "paradoxical" indentity-based signature scheme resulting from zero-knowledge. In *Advances in Cryptology—CRYPTO'88: Proceedings 8* (pp. 216-231). Springer New York.

Işıklı, B. (2022). Eliptik eğrilerin şifrelemede kullanımı (Yüksek lisans tezi, Erzincan Binali Yıldırım Üniversitesi). Yükseköğretim Kurulu Ulusal Tez Merkezi.

Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, *48*(177), 203–209.

Koç, Ç. K., Özdemir, F. and Özger, Z. Ö. (2021), "Partially Homomorphic Encryption" Springer.

Kurtaran, E. (1999). *Digital Signature Schemes in Public-Key Cryptosystems* (Master's thesis, Marmara Universitesi (Turkey)).

Lamport, L. (1979). Constructing digital signatures from a one way function.

Menezes, A., Vanstone, S., Van Oorschot, P. (1996). Handbook of Applied Cryptography. CRC Press

Merkle, R. C. (1979). *Secrecy, authentication, and public key systems* (Technical Report No. 1979-1-6). Stanford University, Electrical Engineering Department.

Miller, V. S. (1986). Use of elliptic curves in cryptography. *Advances in Cryptology—CRYPTO '85 Proceedings*, 417, 417–426.

National Institute of Standards and Technology. (2020). *Recommendation for key management: Part 1 – General* (NIST Special Publication 800-57 Part 1 Rev. 5). U.S. Department of Commerce. https://doi.org/10.6028/NIST.SP.800-57pt1r5

National Institute of Standards and Technology. (2022). *Post-Quantum Cryptography Standardization: Finalist Report*. U.S. Department of Commerce. https://csrc.nist.gov/publications/detail/nistir/8309/final

Nyberg, K., & Rueppel, R. A. (1993, December). A new signature scheme based on the DSA giving message recovery. In *Proceedings of the 1st ACM Conference on Computer and Communications Security* (pp. 58-61).

Okumuş, İ. (2012) "RSA Kriptosisteminin Hızını Etkileyen Faktörler", Doktora Tezi, Atatürk Üniversitesi Fen Bilimleri Enstitüsü, Erzurum.

PUB, F. (1998). Digital signature standard (DSS). *Fips pub*, 186-191.

PUB, F. (2000). Digital signature standard (DSS). *Fips pub*, 186-192.

PUB, F. (2009). Digital signature standard (DSS). *Fips pub*, 186-193.

PUB, F. (2013). Digital signature standard (DSS). *Fips pub*, 186-194.

PUB, F. (2023). Digital signature standard (DSS). *Fips pub*, 186-195.

Rabin, M. O. (1979) "Digitalized Signatures and Public-Key Functions as Intractable as Factorization", Massachusetts Institute of Technology, USA.

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

Rosen, K. H. (1984) Elementary number theory and lts Applications, Addison-Wesley Publishing Company.

Robert, W. (1979) "A Secure One-Way Hash Function Built from DES," *IBM Technica Disclosure Bulletin*, Vol. 27, No. 11A.

Rosen, K. H. (2015). Ayrık Matematik ve Uygulamaları 7. Baskıdan Çeviri, Prof. Dr. Ömer Akın, Yrd. Doç. Dr. Murat Özbayoğlu, Palme Yayıncılık, Ankara, 204 214.

Schnorr, C. P. (1990). Factoring Integers and Computing Discrete Logarithms via Diophantine Approximation. In *Advances in Computational Complexity Theory* (pp. 171-181).

Schnorr, C. P. (1990). Efficient identification and signatures for smart cards. In *Advances in Cryptology—CRYPTO'89 Proceedings 9* (pp. 239-252). Springer New York.

Schnorr, C. P. (1991). Efficient signature generation for smart cards. *Journal of Cryptology*, *4*(3), 161–174.

Soyalıç, S. (2005). Kriptografik Hash fonksiyonları ve uygulamaları (Yüksek lisans tezi, Erciyes Üniversitesi). Yükseköğretim kurulu Ulusal Tez Merkezi.

Taşcı, D. (2007) Soyut Cebir, Alp Yayınevi, Ankara.

Winternitz, H. (1982). *A new digital signature scheme based on the hardness of computing square roots modulo N* (Technical Report). MIT Laboratory for Computer Science.

ÖZGEÇMİŞ

01 Kasım 1999 tarihinde Erzincan'ın Çayırlı ilçesinde doğdum. Lise eğitimimi 2013–2017 yılları arasında Erzincan Atatürk Mesleki ve Teknik Anadolu Lisesi'nde tamamladım. Lisans eğitimime 2018 yılında Erzincan Binali Yıldırım Üniversitesi Fen-Edebiyat Fakültesi Matematik Bölümü'nde başladım ve 2022 yılında mezun oldum. 2023 yılında aynı üniversitenin Matematik Anabilim Dalı'nda yüksek lisans eğitimime başladım.