ELİPTİK EĞRİLER İLE SAYISAL İMZA

Fatih DEĞİRMENCİ

YÜKSEK LİSANS TEZİ
ELEKTRİK – ELEKTRONİK MÜHENDİSLİĞİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ARALIK 2006
ANKARA

Fatih DEĞİRMENCİ tarafından hazırlanan ELİPTİK EĞRİLER İLE SAYISAL İMZA adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Erkan AFACAN
Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği ile Elektrik – Elektronik Mühendisliği Anabilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Başkan: : Prof. Dr. Erdem YAZGAN

Üye : Yrd. Doç. Dr. Erkan AFACAN

Üye : Yrd. Doç. Dr. K. Cem NAKİBOĞLU

Tarih : 28/12/2006

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Fatih DEĞİRMENCİ

ELİPTİK EĞRİLER İLE SAYISAL İMZA

(Yüksek Lisans Tezi)

Fatih DEĞİRMENCİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Aralık 2006

ÖZET

Bu tezde, Açık Anahtarlı Şifreleme Sistemleri ve bu sistemlerin gerçekleştirilmesinde temel olan matematiksel özellikler incelenmiş ve özellikle Eliptik Eğri Sayısal İmza Sistemi'nin simülasyonu gerçekleştirilmiştir. Sayısal imza oluşturma algoritması detaylı bir şekilde ele alınmış ve diğer bir açık anahtarlı şifreleme sistemi olan RSA ile olan farklılıklar ortaya konulmaya çalışılmıştır. Eliptik Eğri Sayısal İmza Sistemi'nin simülasyonu için C++ ve Nesneye Yönelik Programlama Teknikleri kullanılarak bir uygulama geliştirilmiştir. Geliştirilmiş olan uygulama yardımıyla sayısal imza oluşturma işlemi modellenmiştir.

Bilim Kodu : **905.1.011**
Anahtar Kelimeler : **Kriptografi, Eliptik eğriler, Sayısal imza, RSA**
Sayfa Adedi : **76**
Tez Yöneticisi : **Yrd. Doç. Dr. Erkan AFACAN**

DIGITAL SIGNATURES WITH ELLIPTIC CURVES

(M.Sc. Thesis)

Fatih DEĞİRMENCİ

GAZI UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

December 2006

ABSTRACT

In this thesis, Public Key Cryptography and mathematical foundations are examined. Simulation of Elliptic Curve Digital Signature System are realized. Another Public Key Cryptosystem RSA is also covered. Digital Signature Generation Algorithm is explained in detail and differences between RSA and Elliptic Curve Cryptographic Systems are examined. For the simulation of the Elliptic Curve Digital Signature System, an application is developed using C++ and Object Oriented Programming Techniques. Digital signature generation process is modelled with this application.

Science Code : **905.1.011**
Key Words : **Cryptography, Elliptic curves, Digital signatures, RSA**
Page Number : **76**
Adviser : **Asst. Prof. Dr. Erkan AFACAN**

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren Sayın Hocam Yrd. Doç. Dr. Erkan AFACAN'a ve çalışmalarımda beni hep destekleyen değerli aileme teşekkürü bir borç bilirim.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

1. GİRİŞ

Teknolojinin hızla ilerlemekte olduğu günümüz koşullarında insan hayatını kolaylaştırmaya yönelik birçok yenilik ortaya çıkmış ve gündelik hayat içerisinde büyük öneme haiz işlemler bu yenilikler yardımı ile kolaylıkla gerçekleştirilmeye başlanmıştır. Ortaya konulan bu yenilikler, çeşitli güvenlik problemlerini de beraberinde getirmiştir.

Haberleşme ortamları kullanılarak yapılan birçok işlemde güvenlik konusu yıllar ilerledikçe daha önemli hale gelmiş ve bu tarz problemlerin aşılmasına yönelik birçok metot ortaya koyulmaya çalışılmıştır. Özellikle, bilgisayar ve internet tabanlı haberleşme sistemlerinde şifreleme, şifre çözme, bilgi kaynağının doğrulanması ve bu gibi birçok konu üzerinde çeşitli çalışmalar yapılmış ve yapılmaya devam etmektedir.

Gereksinimler sonucu ortaya çıkmış olan en önemli şifreleme sistemleri Açık Anahtarlı Şifreleme Sistemleri'dir. Günümüzde en sık kullanılan sistemlerden RSA Şifreleme Sistemi ve Eliptik Eğri Şifreleme Sistemi de birer Açık Anahtarlı Şifreleme Sistemi'dir.

Bu tez kapsamında Açık Anahtarlı Şifreleme Sistemleri ve özellikle Eliptik Eğri Şifreleme Sistemleri incelenmiş olup, Eliptik Eğri Sayısal İmza Algoritması üzerinde durulmuştur. Algoritmanın gerçekleştirilmesi için gerekli olan diğer temel konulara değinilmiş olup, elde edilen neticeler sonucunda Eliptik Eğri Sayısal İmza Algoritması, geliştirilmiş olan bir program yardımı ile modellenmiştir.

İkinci bölümde, şifreleme sistemlerinin temelinde yer alan matematiksel kavramlar, asal sayılar ve asallık testleri incelenmiş ve çeşitli örnekler verilmiştir.

Üçüncü bölümde, temel iletişim modeli ve gereksinimler ortaya konulmuştur. Bu gereksinimlerin karşılanmasına yönelik şifreleme sistemlerine değinilerek kriptografinin temel özelliklerinden bahsedilmiştir.

Dördüncü bölümde, Açık Anahtarlı Şifreleme Sistemleri incelenmektedir. Bir diğer Açık Anahtarlı Şifreleme Sistemi olan RSA Şifreleme Sistemine de bu bölümde kısaca değinilmiş olup bu sistemlerin belirgin özellikleri ve sağladığı avantajlar açıklanmıştır.

Beşinci bölümde, eliptik eğrilerin matematiksel özellikleri derinlemesine incelenmiş olup, eliptik eğrilerin kriptografideki kullanım alanlarına göre çeşitli fonksiyonlarına da değinilmiştir.

Altıncı bölümde, Açık Anahtarlı Şifreleme Sistemleri'nin oluşturulmasında büyük öneme sahip olan ayrık logaritma problemi, eliptik eğriler perspektifinde incelenmiştir.

Yedinci bölümde, sayısal imzalar incelenmiş olup, sayısal imzalar ile geleneksel imzalar arasındaki farklar ve eliptik eğri sayısal imza oluşturma algoritması örnekler yardımıyla açıklanmıştır.

Son bölümde, eliptik eğri sayısal imza uygulaması ile ilgili yazılım özelliklerine değinilerek program arayüzüne yer verilmiş olup programın kullanımına yönelik çeşitli konular açıklanmıştır.

2. MATEMATİKSEL KAVRAMLAR

Bu bölümde aritmetik ile ilgili temel kavramlar açıklanmış olup özellikle de sayılar ve sayılar üzerinde ortaya konulmuş olan çeşitli teoremlere değinilmiştir.

2.1. Gruplar

2.1 Tanım

Üzerinde tersi alınabilir ve birleşme özelliğine sahip iki değişkenli bir operasyonun tanımlı olduğu kümeler *grup* olarak adlandırılır. Grup, boş olmayan bir G kümesi ve onun üzerinde tanımlı bir $\cdot$ işleminden oluşur. Bu operasyonun aşağıdaki şartları sağlaması gereklidir;

i) $\cdot$ işlemi, G'de *kapalıdır*.

ii) $\cdot$ işlemi, $\forall a,b,c \in G$ için *birleşme özelliğini* sağlar.

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

iii) $\cdot$ işleminin $e \in G$ şeklinde bir *birim elemanı* vardır.

$$a \cdot e = e \cdot a = a$$

iv) $\cdot$ işlemine göre G'deki her elemanın *tersi* vardır.

$$a \cdot a^{-1} = a^{-1} \cdot a = e$$

$(G,\cdot)$ yapısı sadece i ve ii aksiyomlarını sağlıyorsa bir *yarı grup* denir.

$(G,\cdot)$ yapısı bir grupsa ve $\cdot$ işleminin değişme özelliği de varsa bir *değişmeli grup* denir.

2.2. Halka ve Cisimler

2.2 Tanım

R boş olmayan bir küme olsun. + ve · ikili işlemleri bu küme üzerinde tanımlı olsunlar. Eğer $(R,+,\cdot)$ cebirsel yapısı aşağıdaki aksiyomları sağlıyorsa bu cebirsel yapıya *halka* adı verilir.

i) $(R,+)$ değişmeli gruptur.
ii) · işleminin *R*'de birleşme özelliği vardır.
iii) · işleminin + işlemi üzerine sağdan ve soldan dağılma özelliği vardır.

Bir halkanın + (toplama) işlemine göre etkisiz elemanına *sıfır eleman* denir. Halkanın · (çarpma) işlemine göre etkisiz elemanı varsa bu halkaya *birimli halka* denir. Ayrıca · işlemine göre değişmeli ise bu halkaya *birimli ve değişmeli halka* denir.

2.3 Tanım

R, birimli ve değişmeli bir halka ve $R' = R - \{O_R\}$ ikinci işlem ·'a göre bir grup ise *R*'ye *cisim* denir. Bir cisimde sıfırdan farklı her elemanın çarpımsal tersi vardır ve tektir [10-17].

2.3. Tamsayılar ve Tamsayılarda Aritmetik

..., -3, -2, -1, 0, 1, 2, 3, ... sayılarına *tamsayılar*, 0, 1, 2, ... sayılarına *negatif olmayan tamsayılar* ve 1, 2, 3, ... sayılarına da *pozitif tamsayılar* denir. Tamsayıların oluşturduğu kümeye *tamsayılar kümesi* denir ve $\mathbb{Z}$ ile gösterilir [7].

Tamsayıların şu özellikleri vardır; $(a,b,c \in \mathbb{Z}$ için)

i) Kapalılık özelliği: $a+b \in \mathbb{Z}$ ve $a \cdot b \in \mathbb{Z}$.
ii) Değişme özelliği: $a+b=b+a$ ve $a \cdot b = b \cdot a$.

iii) Birleşme özelliği: $a+(b+c)=(a+b)+c$ ve $a\cdot(b\cdot c)=(a\cdot b)\cdot c$.

iv) Dağılma özelliği: $a\cdot(b+c)=a\cdot b+a\cdot c$ ve $(b+c)\cdot a=b\cdot a+c\cdot a$.

v) Kısaltma özelliği: $a+c=b+c$ ise $a=b$ ve $a\cdot c=b\cdot c$ ise $a=b$. $(c\neq 0)$

vi) Üç hal kuralı: $a=b, a<b, a>b$ ifadelerinden yalnızca biri doğrudur.

2.3.1. Bölünebilme

2.4 Tanım

$a,b\in\mathbb{Z}$ için $b=a\cdot c$ olacak şekilde bir $c\in\mathbb{Z}$ bulunabilirse, a,b'yi böler denir ve $a\mid b$ ile gösterilir. Bu c tamsayısını bulmaya *b'yi a ile bölme işlemi* denir. *b'*ye *bölünen*, *a*'ya *bölen* ve *c*'ye *bölüm* denir. *a* ve *c*'ye *b'nin bölenleri* denir.

$a,b,c,d,m,n\in\mathbb{Z}$ olmak üzere bölünebilmenin aşağıdaki özellikleri vardır:

i) Yansıma özelliği: $a \mid a$.

ii) Geçişme özelliği: $a \mid b$ ve $b \mid c$ ise $a \mid c$.

iii) Doğrusallık özelliği: $a \mid b$ ve $a \mid c$ ise $x, y\in\mathbb{Z}$ olmak üzere $a \mid b\cdot x+c\cdot y$.

iv) $a \mid b$ ise $a \mid b\cdot c$.

v) $m\neq 0$ ve $a \mid b \Leftrightarrow m\cdot a \mid m\cdot b$.

vi) $1 \mid a$.

vii) $a \mid 0$.

viii) $0 \mid n$ ise $n=0$.

ix) $a \mid b$ ve $b \mid a$ ise $a=\pm b$.

x) $a \mid b$ ve $a>0, b>0$ $a\leq b$.

xi) $a \mid b$ ve $a\neq 0$ ise $\frac{b}{a} \mid b$.

2.1 Teorem (Kalanlı bölme teoremi)

Herhangi $a > 0$ ve b tamsayıları için $b = a \cdot q + r$ ve $0 \le r < a$ olacak şekilde tek türlü belirli q, r tamsayı çifti vardır. Eğer $a \mid b$ sağlanmıyor ise r tamsayısı daha kuvvetli olan $0 < r < a$ eşitsizliğini sağlar [10].

2.3.2. Euclid algoritması

2.2 Teorem (Euclid teoremi)

Asal sayılar sonsuz çokluktadır.

2.3 Teorem (Euclid algoritması)

Ardarda yapılan kalanlı bölmeler arasında 0'dan farklı en son kalana m ve n'in *en büyük ortak böleni* denir ve $d = (m,n)$ ile gösterilir.

$$
\begin{aligned}
n &= q_1 \cdot m + r_1, && (0 \le r_1 < m)\\
m &= q_2 \cdot r_1 + r_2, && (0 \le r_2 < r_1)\\
r_1 &= q_3 \cdot r_2 + r_3, && (0 \le r_3 < r_2)\\
&\ \ \vdots\\
r_{k-1} &= q_{k+1} \cdot r_k + r_{k+1}, && (0 \le r_{k+1} < r_k)\\
r_k &= q_{k+2} \cdot r_{k+1} + 0
\end{aligned}
$$

Yapılan işlemler sonucunda $obeb(m,n) = r_{k+1}$ olarak bulunur.

2.1 Örnek

1817 ile 1201'in en büyük ortak böleni Euclid Algoritması ile hesaplanacak olursa;

$$
\begin{aligned}
1817 &= 1\cdot 1201 + 616\\
1201 &= 1\cdot 616 + 585\\
616 &= 1\cdot 585 + 31\\
585 &= 18\cdot 31 + 27\\
31 &= 1\cdot 27 + 4\\
27 &= 6\cdot 4 + 3\\
4 &= 1\cdot 3 + 1\\
3 &= 3\cdot 1 + 0
\end{aligned}
$$

elde edilir.

Yapılan işlemler sonucunda $obeb(1817,1201)=1$ olduğu görülür.

Yukarıdaki yöntem kullanılarak $d = x\cdot m + y\cdot n$ olacak şekilde $x, y \in \mathbb{Z}$ tamsayıları bulunabilir.

$$
\begin{aligned}
1 &= x\cdot 1817 + y.1201\\
1 &= 4 - 1\cdot 3\\
1 &= 4 - 1\cdot(27 - 6\cdot 4) = 7\cdot 4 - 1\cdot 27\\
1 &= 7\cdot(31 - 1\cdot 27) - 1\cdot 27 = 7\cdot 31 - 8\cdot 27\\
1 &= 7\cdot 31 - 8\cdot(585 - 18\cdot 31 == 151\cdot 31 - 8\cdot 585\\
1 &= 151\cdot(616 - 1\cdot 585) - 8\cdot 585 = 151\cdot 616 - 159\cdot 585\\
1 &= 151\cdot 616 - 159\cdot(1201 - 1\cdot 616) = 310\cdot 616 - 159\cdot 1201\\
1 &= 310\cdot(1817 - 1\cdot 1201) - 159\cdot 1201 = 310\cdot 1817 - 469\cdot 1201\\
1 &= 310\cdot 1817 - 469\cdot 1201
\end{aligned}
$$

Bulunan tamsayılar $x = 310$ ve $y = -469$ şeklindedir [10].

2.3.3. Aritmetiğin temel teoremi

Her $a > 1$ tamsayısının asal çarpanlarına ayrılışı, sıra düşünmeksizin tek türlüdür.

2.2 Örnek

$$300 = 2^2\cdot 3\cdot 5^2$$

2.5 Tanım

m ve *n* sıfırdan farklı iki tamsayı olsun;

i) $m \mid k$ ve $n \mid k$ olacak şekilde bir $k > 0$ tamsayısı varsa *k*'ya *m* ile *n*'in *bir ortak katı* denir.
ii) *k*, *m* ile *n*'in bir ortak katı olsun. Eğer *m* ile *n*'in her ortak katı için $k \mid t$ ise, *k*'ya *m* ile *n*'in *en küçük ortak katı* denir ve $k = [m, n]$ ile gösterilir.

$$a = p_1^{m_1} \cdot p_2^{m_2} \cdot \ldots \cdot p_r^{m_r}$$

ve

$$b = p_1^{n_1} \cdot p_2^{n_2} \cdot \ldots \cdot p_r^{n_r}$$

olsun.

$$(a,b) = p_1^{\min(m_1,n_1)} \cdot p_2^{\min(m_2,n_2)} \cdot \ldots \cdot p_r^{\min(m_r,n_r)} \rightarrow obeb$$
$$[a,b] = p_1^{\max(m_1,n_1)} \cdot p_2^{\max(m_2,n_2)} \cdot \ldots \cdot p_r^{\max(m_r,n_r)} \rightarrow okek$$

Önerme

a ve *b* pozitif tamsayılar olsun. $(a,b) \cdot [a,b] = a \cdot b$.

2.3.4. Euler fonksiyonu

2.6 Tanım

Pozitif *n* tamsayısı için $1 \le a \le n$ ve $(n,a) = 1$ olan *a* tamsayılarının sayısı $\phi(n)$ ile gösterilir ve *Euler Fonksiyonu* denir.

$n = 1 \quad (1,1) = 1 \quad \phi(1) = 1 \quad 1 \le a \le 1$

$n = 2 \quad (2,1) = 2 \quad \phi(2) = 2 \quad 1 \le a \le 2$

$(2,2) = 2$

$n = 3 \quad (3,1) = 1 \quad \phi(3) = 2 \quad 1 \le a \le 3$

$(3,2) = 1$

$(3,3) = 3$

$n = 4 \quad (4,1) = 1 \quad \phi(4) = 2 \quad 1 \le a \le 4$

$(4,2) = 2$

$(4,3) = 1$

$(4,4) = 4$

$n = 5 \quad (5,1) = 1 \quad \phi(5) = 4 \quad 1 \le a \le 5$

$(5,2) = 1$

$(5,3) = 1$

$(5,4) = 1$

$(5,5) = 5$

$n = 6 \quad (6,1) = 1 \quad \phi(6) = 2 \quad 1 \le a \le 6$

$(6,2) = 2$

$(6,3) = 3$

$(6,4) = 2$

$(6,5) = 1$

$(6,6) = 6$

$n = 7 \quad (7,1) = 1 \quad \phi(7) = 6 \quad 1 \le a \le 7$

$(7,2) = 1$

$(7,3) = 1$

$(7,4) = 1$

$(7,5) = 1$

$(7,6) = 1$

$(7,7) = 7$

Euler Fonksiyonu'nun özelliklerine göz atılacak olursa;

i) p asal sayı ise,

$$\phi(p) = p-1 = p\cdot\left(1-\frac{1}{p}\right),\ \phi(2) = 2-1 = 2\cdot\left(1-\frac{1}{2}\right)=1$$
$$\phi(3) = 3-1 = 3\cdot\left(1-\frac{1}{3}\right)=1,\ \phi(7) = 7-1 = 7\cdot\left(1-\frac{1}{7}\right)=1$$

ii) p asal sayı ve $a\in\mathbb{N}$ ise,

$$\phi(p^a) = p^a - p^{a-1} = p^a\cdot\left(1-\frac{1}{p}\right)$$
$$\phi(4) = 2^2 - 2^1 = 2$$
$$\phi(8) = 2^3 - 2^2 = 4$$
$$\phi(9) = 3^2 - 3^1 = 6$$

iii) $(m,n)=1$ ise,

$$\phi(m,n) = \phi(m)\cdot\phi(n)$$
$$\phi(10) = \phi(5)\cdot\phi(2) = 4\cdot 1 = 4$$

iv) $m = p_1^{a_1}\cdot p_2^{a_2}\cdot\ldots\cdot p_r^{a_r}$

$$\phi(m) = p_1^{a_1}\cdot p_2^{a_2}\cdot\ldots\cdot\left(1-\frac{1}{p_1}\right)\cdot\left(1-\frac{1}{p_2}\right)\cdot\ldots\cdot\left(1-\frac{1}{p_r}\right)$$
$$\phi(m) = m\cdot\left(1-\frac{1}{p_1}\right)\cdot\left(1-\frac{1}{p_2}\right)\cdot\ldots\cdot\left(1-\frac{1}{p_r}\right)$$
$$\phi(144) = \phi(2^4\cdot 3^2) = 2^4\cdot 3^2\cdot\left(1-\frac{1}{2}\right)\cdot\left(1-\frac{1}{3}\right)=48$$

2.3.5. Modüler aritmetik

2.7 Tanım

m sıfırdan farklı bir tamsayı olsun. $a,b\in\mathbb{Z}$ için;

$a \equiv b \pmod{m} \Rightarrow m \mid a-b$

ile tanımlanır.

Önerme

Yukarıda tanımlanan $\equiv$ bağıntısı bir *denklik bağıntısı*dır.

i) Yansıma: $a \equiv a$.
ii) Simetri: $a \equiv b$ ise $b \equiv a$.
iii) Geçişme: $a \equiv b$, $b \equiv c$ ise $a \equiv c$.

2.8 Tanım

$\mathbb{Z}$'deki denklik bağıntısının belirttiği denklik sınıflarına, m modülüne göre (mod m) *kalan sınıfları* denir ve tüm kalan sınıfları kümesi $\mathbb{Z}_m$ ile gösterilir. $a \in \mathbb{Z}$'nin denklik sınıfı;

$\overline{a} = \{x \in \mathbb{Z} : m \mid a-x\}$

Önerme

$a \equiv b \pmod{m} \Leftrightarrow a$ ve b'nin m ile bölümünden elde edilen kalanın aynı olmasıdır.

Not: Bir $a \in \mathbb{Z}$'nin $m > 0$ ile bölümünden elde edilen kalanlar $0, 1, 2, \ldots, m-1$ olacağından $\overline{a}$ sınıfı, $\overline{0}, \overline{1}, \overline{2}, \ldots, \overline{m-1}$ sınıflarından biridir. $\mathbb{Z}_m$, m elemanlı bir kümedir.

2.9 Tanım

$\mathbb{Z}_m$'de her sınıftan bir ve yalnız bir eleman almakla elde edilen sisteme *tam temsilciler sistemi* denir. $\mathbb{Z}_m$'nin bir tam temsilciler sistemi olarak {0, 1, 2,…} alınabilir.

2.3 Örnek

$\mathbb{Z}_6 = \{\overline{0},\ \overline{1},\ \overline{2},\ \overline{3}, \overline{4}, \overline{5}\}$

$14 \equiv 2 \pmod 6,\ 14 \in \overline{2}$
$23 \equiv 5 \pmod 6,\ 23 \in \overline{5}$

$\overline{0} = \{\ldots, -12, -6, 0, 6, 12, \ldots\}$
$\overline{1} = \{\ldots, -11, -5, 1, 7, 13, \ldots\}$
$\overline{2} = \{\ldots, -10, -4, 2, 8, 14, \ldots\}$
$\overline{3} = \{\ldots, -9, -3, 3, 9, 15, \ldots\}$
$\overline{4} = \{\ldots, -8, -2, 4, 10, 16, \ldots\}$
$\overline{5} = \{\ldots, -7, -1, 5, 11, 17, \ldots\}$

Önerme

$a \equiv a_1 \pmod m$ ve $b \equiv b_1 \pmod m$ ise $a + b \equiv a_1 + b_1 \pmod m$ olur.

Önerme

$a \equiv a_1 \pmod m$ ve $b \equiv b_1 \pmod m$ ise $a \cdot b \equiv a_1 \cdot b_1 \pmod m$ olur.

2.4 Örnek

$\mathbb{Z}_6$'da;

$\overline{2} \oplus \overline{5} = \overline{7} \equiv \overline{1}$
$\overline{2} \otimes \overline{5} = \overline{10} \equiv \overline{4}$

Önerme

$\forall \overline{a}, \overline{b}, \overline{c} \in \mathbb{Z}_m$ için $\mathbb{Z}_m$'de $\oplus$ işleminin şu özellikleri vardır;

i) Değişme: $\overline{a} \oplus \overline{b} = \overline{b} \oplus \overline{a}$.

ii) Birleşme: $\overline{a} \oplus (\overline{b} \oplus \overline{c}) = (\overline{a} \oplus \overline{b}) \oplus \overline{c}$.

iii) Etkisiz eleman: $\overline{a} \oplus \overline{0} = \overline{a}$.

iv) Ters eleman: $\overline{a} \oplus \overline{x} = \overline{0}$ olacak şekilde $\exists \overline{x} \in \mathbb{Z}_m$ bulunabilir.

Önerme

$\forall \overline{a}, \overline{b}, \overline{c} \in \mathbb{Z}_m$ için $\mathbb{Z}_m$'de $\otimes$ işleminin şu özellikleri vardır;

i) Değişme: $\overline{a} \otimes \overline{b} = \overline{b} \otimes \overline{a}$.

ii) Birleşme: $\overline{a} \otimes (\overline{b} \otimes \overline{c}) = (\overline{a} \otimes \overline{b}) \otimes \overline{c}$.

iii) Birim eleman: $\overline{a} \otimes \overline{1} = \overline{a}$.

iv) Yutan eleman: $\overline{a} \otimes \overline{0} = \overline{0}$.

Önerme

$\forall \overline{a}, \overline{b}, \overline{c} \in \mathbb{Z}_m$ için;

$\overline{a} \otimes (\overline{b} \oplus \overline{c}) = (\overline{a} \otimes \overline{b}) \oplus (\overline{a} \otimes \overline{c})$.

olur. ($\otimes$'nın $\oplus$ üzerine dağılma özelliği) $\mathbb{Z}_6$'da $\overline{2} \otimes \overline{3} = \overline{6} = \overline{0}$ eşitliğinde olduğu gibi sıfır olmayan iki sınıfın çarpımı sıfır olabilir. $\mathbb{Z}$'de ise bu mümkün değildir.

2.10 Tanım

$\mathbb{Z}_m$'de kendileri $\overline{0}$'dan farklı olduğu halde çarpımları $\overline{0}$ olan sınıflara *sıfır bölen sınıflar* denir.

2.5 Örnek

$\mathbb{Z}_6$'da sıfır bölen sınıflar $\overline{2},\ \overline{3}, \overline{4}$'tür. $(\overline{2} \otimes \overline{3} \equiv \overline{0},\ \overline{3} \otimes \overline{4} \equiv \overline{0})$

Önerme

$\overline{a} \equiv \overline{b} \pmod{m} \Rightarrow (a,\ m) = (b,\ m)$ olur.

2.11 Tanım

$\overline{a} \in \mathbb{Z}_m$ için $(a, m) = 1$. $\overline{a}$ sınıfına bir *asal kalan sınıfı* denir. $\mathbb{Z}_m$'in bütün asal kalan sınıfları $\mathbb{Z}_m^*$ ile gösterilir.

2.6 Örnek

$\mathbb{Z}_6^* = \{\overline{1}, \overline{5}\},\ \ \phi(6) = \phi(2) \cdot \phi(3) = 1 \cdot 2 = 2$

$\mathbb{Z}_m^*$ kümesinin eleman sayısı $\phi(m)$ Euler Fonksiyonu ile verilir. m modülü asal ise $\mathbb{Z}_m^* = \mathbb{Z}_m - \{0\}$ olur.

Önerme

İki asal kalan sınıfının çarpımı da bir asal kalan sınıfıdır.

Önerme

$\mathbb{Z}_m$'deki $\overline{0}$'dan farklı bir kalan sınıfının sıfır bölen olması için gerek ve yeter koşul asal kalan sınıfı olmamasıdır.

2.12 Tanım

$\overline{a} \in \mathbb{Z}_m$ olsun. $\overline{a} \cdot \overline{c} = 1$ olacak şekilde $\exists \overline{c} \in \mathbb{Z}_m$ varsa, $\overline{c}$ 'ye $\overline{a}$ 'nın tersi denir.

Önerme

$\mathbb{Z}_m$'deki bir kalan sınıfının tersinin olması için gerek ve yeter koşul bir asal kalan sınıfı olmasıdır.

Sonuç

m asal tamsayı ise, $\mathbb{Z}_m$'deki sıfırdan farklı her kalan sınıfının tersi mevcuttur.

2.7 Örnek

$\mathbb{Z}_7 = \{\overline{0}, \overline{1}, \overline{2}, \overline{3}, \overline{4}, \overline{5}, \overline{6}\}$

$$\overline{1} \cdot \overline{1} \equiv \overline{1}$$
$$\overline{2} \cdot \overline{4} \equiv \overline{1}$$
$$\overline{3} \cdot \overline{5} \equiv \overline{1}$$
$$\overline{6} \cdot \overline{6} \equiv \overline{1}$$

2.8 Örnek

$\mathbb{Z}_{10}$'daki asal kalan sınıfları veya tersi mevcut sınıflar $\phi(10) = \phi(5) \cdot \phi(2) = 4 \cdot 1 = 4$

tane olup;

$$\left.\begin{array}{rcl} \overline{1}\cdot\overline{1} & \equiv & \overline{1} \\ \overline{3}\cdot\overline{7} & \equiv & \overline{1} \\ \overline{9}\cdot\overline{9} & \equiv & \overline{1} \end{array}\right\} = \mathbb{Z}_{10}^{*} = \{\overline{1},\ \overline{3},\ \overline{7},\ \overline{9}\}.$$

şeklindedir.

$\mathbb{Z}_{10}$'daki sıfır bölenler de $\overline{2},\ \overline{4},\ \overline{5},\ \overline{6},\ \overline{8}$ 'dir. $(\overline{2}\cdot\overline{5}\equiv\overline{0},\ \overline{4}\cdot\overline{5}\equiv\overline{0})$

2.3.6. Euler teoremi

2.4 Teorem

$m\in\mathbb{Z}$ ve $(a,\ m)=1$ olsun. $\forall a\in\mathbb{Z}$ için;

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

veya

$\overline{a}^{\phi(m)}\equiv\overline{1}$.

İspat

$\mathbb{Z}_m^{*}$ asal kalan sınıflar kümesini düşünecek olursak;

$$\mathbb{Z}_m^{*} = \{a\in\mathbb{Z} : 1\le a\le m \text{ ve } (a,\ m)=1\}$$

$$\mathbb{Z}_m^{10} = \{1,\ 3,\ 7,\ 9\}$$

$$\mathbb{Z}_m^{7} = \{1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 7\}$$

$\phi(m)$: $\mathbb{Z}_m^{*}$ kümesinin eleman sayısı

$a \in \mathbb{Z}_m^*$ sabit bir asal kalan sınıfı alacak olursak $\forall \bar{b} \in \mathbb{Z}^m$ için;

$$f(\bar{b}) = \bar{a} \otimes \bar{b}$$

ile $f : \mathbb{Z}_m^* \to \mathbb{Z}_m^*$ fonksiyonu tanımlayalım. $\bar{a} \otimes \bar{b} \in \mathbb{Z}_m^*$ olur. f'in bire bir olduğunu gösterelim;

$$f(\bar{b}) = f(\bar{c}) \Rightarrow \bar{a} \otimes \bar{b} = \bar{a} \otimes \bar{c} \Rightarrow \bar{b} = \bar{c}$$

elde edilir.

$\mathbb{Z}_m^*$ sonlu elemanlı bir küme olduğundan f'in bire bir olması, örten olmasını da gerektirir. Şu halde f, $\mathbb{Z}_m^*$'in elemanlarını aralarında değiştirir. Buradan

$$\mathbb{Z}_m^* = \{\bar{a}_1,\ \bar{a}_2,\ \ldots,\ \bar{a}_{\phi(m)}\}$$
$$f(\bar{a}_1) \cdot f(\bar{a}_1) \cdot \ldots \cdot f(\bar{a}_{\phi(m)}) = \bar{a}_1 \cdot\ \bar{a}_2 \cdot \ldots \cdot \bar{a}_{\phi(m)}$$
$$f(\bar{b}) = \bar{a} \otimes \bar{b}$$
$$f(\bar{a}_1) = \bar{a} \otimes \bar{a}_1$$
$$f(\bar{a}_2) = \bar{a} \otimes \bar{a}_2$$
$$\vdots$$
$$f(\bar{a}_{\phi(n)}) = \bar{a} \otimes \bar{a}_{\phi(m)}$$
$$f(\bar{a}_1 \cdot \bar{a}_2 \cdot \ldots \cdot \bar{a}_{\phi(m)}) = \bar{a}_1 \cdot \bar{a}_2 \cdot \ldots \cdot \bar{a}_{\phi(m)} = \bar{a} \cdot \bar{a}_1 \cdot \bar{a} \cdot \bar{a}_2 \cdot \ldots \cdot \bar{a} \cdot \bar{a}_{\phi(m)}$$
$$= (\bar{a})^{\phi(m)} \cdot \bar{a}_1 \cdot \bar{a}_2 \cdot \ldots \cdot \bar{a}_{\phi(m)}$$

2.9 Örnek

$m = 100$ ve $a = 21$ ise

$$(100,\ 21) = 1, \quad \phi(100) = 100 \cdot \frac{1}{2} \cdot \frac{4}{5} = 40$$

$$a^{\phi(m)} = 21^{40} \equiv 1 \pmod{100}$$

Sonuç. (Fermat Teoremi)

$m = p$ asal tamsayı ise $p \chi a$ olan $\forall a \in \mathbb{Z}$ için

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

olur.

2.10 Örnek

23^{37} sayısının 5 ile bölümünden kalan Fermat Teoremi kullanılarak bulunacak olursa;

$23^4 \equiv 1 \pmod{5}$

$23^{37} = 23^{36} \cdot 23 = (23^4)^9 \cdot 23 \equiv 3 \pmod{5}$

2.13 Tanım

$a^x \equiv b \pmod{m}$ şeklinde bir denkleme *bir bilinmeyenli lineer kongrüans* denir. Bu denklemi sağlayan *x* tamsayılarının kümesine de *kongrüansın çözüm kümesi* denir.

Önerme

$a \cdot x \equiv b \pmod{m}$'in bir çözümü $x_0 \in \mathbb{Z}_m$ ise x_0 sınıfındaki bütün tamsayılar da bir çözümdür.

Önerme

$a \cdot x \equiv b \pmod{m}$'in bir çözümünün olması için gerek ve yeter koşul $(a, m) \mid b$ olmasıdır.

Önerme

$(a, m) = 1$ ise $a \cdot x \equiv b \pmod{m}$'in bir çözümü vardır ve mod m tek bir sınıftır.

Sonuç 1

$a \cdot x \equiv b \pmod{m}$ kongrüansı verilsin. $d = (a, m) \mid b$ ise bu kongrüansın çözümleri mod m, d sınıftır.

Sonuç 2

$a \cdot x + b \cdot y = c$ Diophant Denklemi'nin bir çözümü olması için gerek ve yeter koşul $(a, b) = d \mid c$ olmasıdır. Bu takdirde (x_0, y_0) bir çözüm ise $\forall k \in \mathbb{Z}$ için $x = x_0 + \frac{b}{d} \cdot k$ ve $y = y_0 - \frac{a}{d} \cdot k$ da bir çözümdür.

2.11 Örnek

$$38 \cdot x \equiv 16 \pmod{111} \Leftrightarrow 38 \cdot x - 111 \cdot y = 16$$

i) $(38, 111) = 1$ olduğundan çözüm vardır ve mod 111 tek sınıftır.

$$\begin{aligned} 111 &= 2 \cdot 38 + 35 \\ 38 &= 1 \cdot 35 + 3 \\ 35 &= 11 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1 \quad * \\ 2 &= 1 \cdot 2 + 0 \end{aligned}$$

ii)

$$\begin{aligned} 1 &= 3 - 1 \cdot 2 \\ 1 &= 3 - 1 \cdot (35 - 11 \cdot 3) = 12 \cdot 3 - 1 \cdot 35 \\ 1 &= 12 \cdot 38 - 13 \cdot 35 \\ 1 &= 38 \cdot 38 - 13 \cdot 111 \end{aligned}$$

iii)

$$16 = (16\cdot 38)\cdot 38 - (13\cdot 16)\cdot 111$$
$$x = 16\cdot 38$$
$$y = 16\cdot 13$$

Verilen kongrüansın çözümü $\bar{x} = \overline{608} = \overline{53}$'tür.

Önerme(Çinlilerin Kalan Teoremi)

$m, n \in \mathbb{Z}$ ve $d = (m, n)$ olsun.

$x \equiv a \pmod{m}$

$x \equiv b \pmod{n}$

denklem sisteminin çözümü vardır ve bu çözüm mod $m\cdot n$ tektir.

2.12 Örnek

$$x \equiv 2 \pmod 5$$
$$x \equiv 3 \pmod{11}$$
$$x \equiv 2 \pmod 5 \Rightarrow x = 2 + 5\cdot k,\ k \in \mathbb{Z}$$
$$2 + 5\cdot k \equiv 3 \pmod{11}$$
$$5\cdot k \equiv 1 \pmod{11}$$
$$9\cdot 5\cdot k \equiv 9 \pmod{11}$$
$$k \equiv 9 \pmod{11}$$
$$x \equiv 3 + 5\cdot 9 = 47$$
$$x \equiv 47 \pmod{55}$$

2.4. Abelyen Gruplar

Şayet elemanların sırası aritmetik işlemin sonucuna etki etmiyorsa, bu aritmetik işlem değişmelidir. Sıradan sayılar ile yapılan toplama ve çarpma işlemleri, değişmeli işlemlere örnek olarak gösterilebilir [10].

$2 \cdot 9 = 9 \cdot 2$
$2 + 9 = 9 + 2$

Çıkarma ve ve bölme işlemleri ise değişmeli işlemler değildirler.

$2 - 9 \neq 9 - 2$
$2/9 \neq 9/2$

Bir grup, şayet üzerinde tanımlı temel işlem değişmeli ise *abelyen grup* olarak adlandırılır. Örneğin, bir toplamsal grup şayet $a + b = b + a$ özelliğini sağlıyor ise abelyen bir gruptur. Bir çarpımsal grup şayet $a \cdot b = b \cdot a$ özelliğini sağlıyor ise abelyen bir gruptur. Toplamsal grup olan Z_n ve çarpımsal grup olan Z_p^* abelyen gruplardır.

2.5. Sonlu Cisimler

2.5.1. Sonlu cisimlere giriş

Cisim kavramı, bilinen sayı sistemlerinin soyutlaması (örneğin rasyonel sayılar $\mathbb{Q}$, gerçel sayılar $\mathbb{R}$ ve kompleks sayılar $\mathbb{C}$) ve bu sistemlerin temel özelliklerinin bir araya getirilmesi ile oluşan bir kavramdır. F grubu ile birlikte toplama (+ ile gösterilir) ve çarpma ($\cdot$ ile gösterilir) öntanımlı iki aritmetik işlemi barındırırlar ve aşağıdaki aritmetik şartları sağlarlar [1].

i) $(F, +)$ abelyen bir gruptur ve toplamsal özdeşlik o ile gösterilir.
ii) $(F \setminus \{0\}, \cdot)$ abelyen bir gruptur ve çarpımsal özdeşlik 1 ile gösterilir.
iii) Dağılma özelliğini sağlar. için $(a + b) \cdot c = a \cdot c + b \cdot c$.

2.5.2. Cisim işlemleri

Bir cisim toplama ve çarpma olmak üzere iki temel işlem içermektedir. Çıkarma işlemi, toplama işlemi kullanılarak açıklanacak olursa; ($a, b \in F$ olmak üzere)

$a - b = a + (-b)$

Burada bulunan $-b$, F'de tektir. $-b$, b'nin negatifi ya da toplamsal tersi olarak adlandırılır. Şöyle ki;

$b + (-b) = 0$

Benzer olarak bölme işlemi de çarpma işlemi kullanılarak açıklanacak olursa; ($a, b \in F$ ve $b \neq 0$ olmak üzere)

$$\frac{a}{b} = a \cdot b^{-1}$$

Burada bulunan b^{-1}, F'de tektir. b^{-1}, b'nin çarpımsal tersi olarak adlandırılır. Şöyle ki;

$b \cdot b^{-1} = 1$

2.6. Asal Cisimler

p'yi asal bir sayı olarak kabul ettiğimizde, $\{0, 1, 2, \ldots, p-1\}$ tamsayıları, toplama ve çarpma işleminin mod p'ye göre yapılacağı, p. dereceden bir sonlu cisim olacaktır. Bu cisim F_p olarak adlandırılır. Herhangi bir a tamsayısının p'ye bölümüyle elde edilen (a mod p), $0 \leq r \leq p-1$ aralığında olan bir tamsayı kalanı r mevcuttur. Bu işlem *mod p'de indirgeme işlemi* olarak adlandırılır. Konuyu bir örnek vererek açıklamamız gerekirse;

i) Toplama: $17 + 20 = 37 \equiv 8 \pmod{29}$.

ii) Çıkarma: $17 - 20 = -3 \equiv 26 \pmod{29}$.

iii) Çarpma: $17 \cdot 20 = 340 \equiv 21 \pmod{29}$.

iv) Çarpmaya göre tersi: $17^{-1} \equiv 12 \pmod{29}$ $(17 \cdot 12 \equiv 1 \pmod{29})$.

2.7. Asal Sayılar

1'den başka pozitif tam sayılar iki sınıfa bölünebilir. Asal sayılar (prime numbers, örneğin 2, 3, 5 ve 7) çarpanlarına ayrılmazlar. Pozitif bölenleri yalnız 1 ve kendisi olan, 1'den büyük tamsayılara *asal sayı* (prime number) denir ve en küçük asal sayı 2'dir. Asal olmayan sayılar (composites, örneğin 4, 6, 8 ve 9) çarpanlarına ayrılabilirler. Asal olmayan sayılar asal çarpanlarının çarpımı olarak bir ve ancak bir yolla ifade edilebilirler. Asal sayıların kendine özgü önemi aritmetiğin temel teoreminden ileri gelmektedir [13].

Tamsayılar arasında asal sayıların dağılımı sorusu matematiğin yüksek eşikli bir problemidir. Asal sayılar serisi detayda büyük düzensizlikler göstermesine rağmen, genel dağılımının kesin terimlerle formüle edilebilen ve matematiksel inceleme konusu yapılabilen belirli düzenlilik özelliklerine sahip olduğu bulunmuştur.

Asal sayılar tablosu incelenecek olursa, geniş kapsamlı bir tablo olduğunu görülür. Asalların, tablonun daha yüksek kısımlarında daha geniş bir dağılım gösterdiği görülmesine rağmen, dağılımlarıyla ilgili hiçbir işaret gözlenemez.

Asal sayıların sayısı sonsuzdur. Yani sonsuz sayıda asal sayı vardır. Euclid tarafından “Elemanlar” adlı onüç ciltlik eserinde ortaya koyulmuştur.

2.7.1. Asallık testleri

Tamsayıların asallıklarının tespiti için kullanılan birçok algoritma mevcuttur. Bu algoritmalar, tamsayılara uygulanmasıyla ortaya çıkan sonuçlara göre sınıflandırılmaktadır.

Bu tez kapsamında incelenen asallık testleri olasılıksal sonuçlar veren testlerdir. Bu tip algoritmalar hızlı sonuç verir, fakat *n* sayısı asal olmasa dahi asal olduğuna dair çıktı verebilirler.
Olasılıksal algoritma tiplerine başlıca iki örnek verilebilir; İlki *Las Vegas Algoritması*'dır ki bu algoritmanın cevap vermeme olasılığı mevcuttur. İkinci

algoritma ise *Monte Carlo Algoritması*'dır. Bu algoritmada ise verilen cevabın doğru olmama olasılığı bulunmaktadır.

Her iki algoritma arasındaki temel fark şu şekilde açıklanabilir; *n* sayısına Las Vegas Algoritması uygulandığında sonuç alınamama olasılığı mevcuttur. Sonuç alınıyorsa da kesinlikle doğrudur. Yine aynı *n* sayısına Monte Carlo Algoritması uygulanacak olursa, her zaman sonuç alınmaktadır. Ancak algoritma sonucunda ortaya çıkan sonuç sayının asal olduğuna dairse, bu sonucun doğru olmama ihtimali vardır. Yine aynı algoritma herhangi bir sayının asal olmadığı sonucu veriyorsa, bu sonuç kesinlikle doğrudur.

Bahsi geçen algoritma tiplerinden Monte Carlo Algoritması'na başlıca iki örnek verilebilir. İlki *Solovay - Strassen Algoritması*'dır. Bu algoritma *evet hükümlü Monte Carlo Algoritması*'dır. Aynı şekilde *Miller - Rabin Algoritması* da Solovay - Strassen Algoritması ile benzer özellikler gösterir. Her iki algoritma arasındaki fark ise Miller - Rabin Testi'nin Solovay - Strassen Testi'ne nazaran daha hızlı ve daha doğru sonuç vermesidir. Bu testin bilinen bir diğer adı da *kuvvetli sözde - asal testi'*dir (Strong pseudo - prime test) [2].

Solovay – Strassen testi.
Girdi. Test edilecek sayı, *n.*
Çıktı. Asal veya *asal değil*.

i) $1 \le a \le n$ şartını sağlayan rasgele a sayısı seçilir.

ii) $x \leftarrow \left(\frac{a}{n}\right)$.

iii) Eğer $x = 0$ ise;

Sonuç: n asal değildir.

iv) $y \leftarrow a^{(n-1)/2} \pmod{n}$.

v) Eğer $x \equiv y \pmod{n}$ ise;

Sonuç: n asaldır.

Aksi takdirde;

Sonuç: n asal değildir.

Miller – Rabin testi.

Girdi. Test edilecek sayı, *n.*

Çıktı. Asal veya *asal değil*.

i) $n-1=2^k \cdot m$ şeklinde yazılır. (*m* tek sayıdır.)

ii) $1 \le a \le n$ şartını sağlayan rasgele *a* sayısı seçilir.

iii) $b \leftarrow a^m \pmod{n}$.

iv) Eğer $b \equiv 1 \pmod{n}$ ise;

Sonuç: n asaldır.

v) $i \leftarrow 0$'dan $k-1$'e kadar;

Eğer $b \equiv -1 \pmod{n}$ ise;

Sonuç: n asaldır.

Aksi takdirde;

$b \leftarrow b^2 \pmod{n}$

vi) *Sonuç*: *n* asal değildir.

Bu çalışma kapsamında geliştirilmiş olan yazılım içerisinde kullanılan test *Miller - Rabin Testi*'dir.

3. KRİPTOGRAFİNİN TEMELLERİ

Bu bölümde kriptografi ile ilgili temel kavramlar açıklanmış olup bazı kriptografik sistemlere değinilmiştir.

3.1. Temel İletişim Modeli

Temel bir iletişim modeli Şekil 3.1'de gösterilmekte olan örnek yardımı ile incelenebilir.

Şekil 3.1 Temel iletişim modeli

A (Ayşe) ve *B* (Bora) güvenli olmayan bir kanal üzerinden haberleşmek isteyen iki kişidir. *E*'nin (Emre) bu haberleşmeye diğer kişilerin bilgisi olmadan dahil olduğunu kabul ediyoruz.

Çeşitli senaryoları ele alacak olursak;

• *A* ve *B*'nin cep telefonu şebekesi üzerinden haberleşmek isteyen iki kişi olduğunu ve *E*'nin de bu konuşmayı art niyetli olarak dinleyen bir unsur olduğunu kabul edebiliriz.

• *a*'yı, *A* kişisinin kullandığı web tarayıcı ve *b*'yi de *B*'nin online satış için kullandığı site olarak kabul edecek olursak, güvenli olmayan haberleşme kanalı *internet* olacaktır. *E* haberleşmeye dahil olarak trafiği takip edebilir ve *A*'nın kredi kartı bilgilerine ulaşarak bunu kötü amaçlar için kullanabilir.

• *A*'nın *B* ile e – mail kullanarak haberleşmek istediğini düşünecek olursak; *E*, bu e – mail trafiğine dışarıdan dahil olarak hem *A* ve hem de *B* ile ilgili gizli bilgilere ulaşabilir.

Yukarıdaki senaryolara dayanarak güvenlik konusunda sağlanması gereken şartları şu şekilde listeleyebiliriz;

i) Gizlilik (*Confidentiality*) : Bilginin izin verilmiş kişiler dışında görüntülenmesinin engellenmesi, gizlenmesi – *A* ve *B* arasındaki haberleşme *E* tarafından görüntülenememelidir.

ii) Bilgi Bütünlüğü (*Data integrity*) : Bilginin yetkisiz kişiler tarafından değiştirilememesinin sağlanması – *B*, *A*'nın kendisine gönderdiği bilginin *E* tarafından değiştirilip değiştirilmediğini tespit edebilmelidir.

iii) Bilgi kaynağının doğrulanması (*Data origin authentication*) : Bilgi kaynağının doğru olup olmadığının tespit edilebilmesi – *B* kendisine gelen mesajın *A* tarafından gönderildiğini doğrulayabilmelidir.

iv) İnkar edememe (*Non-repudiation*) : Kişinin önceki haberleşmelerini reddedememesin sağlanması – *A*, *B*'ye daha önce gönderdiği mesajları inkar edememelidir.

Bazı diğer uygulamalar, kimlik gizlenmesi (*anonymity*) ve erişim kontrolü (*access control*) gibi güvenlik özelliklerini de sağlamaktadır.

3.2. Kriptografi

Güncel kriptografi uygulamalarına geçmeden önce bazı kavramlara ve tarihsel kriptografi sistemlerine göz atmamız yerinde olacaktır.

3.1 Tanım

Kriptografi sistemi aşağıdaki koşulları sağlayan bir ($\mathcal{P}$, $\mathcal{C}$, $\mathcal{K}$, $\mathcal{E}$, $\mathcal{D}$) beşlisidir.

i) $\mathcal{P}$, sonlu elemanlı *düz metinler* (plaintext) kümesidir.

ii) $\mathcal{C}$, sonlu elemanlı *şifreli metinler* (ciphertext) kümesidir.

iii) $\mathcal{K}$, sonlu elemanlı *anahtarlar* (key) kümesidir. (anahtar uzayı)

iv) Her $K \in \mathcal{K}$ için bir $e_{\mathcal{K}} \in \mathcal{E}$ *şifreleme kuralı* (encryption) ve buna karşılık gelen $d_{\mathcal{K}} \in \mathcal{D}$ *şifre çözme kuralı* (decryption) vardır. Her $X \in \mathcal{P}$ düz metni için;

$$d_{\mathcal{K}}(e_{\mathcal{K}}(x)) = \mathcal{X}$$

eşitliği sağlanır.

$$e_{\mathcal{K}} : \mathcal{P} \to \mathcal{C}$$
$$d_{\mathcal{K}} : \mathcal{C} \to \mathcal{P}$$

Bir $\mathcal{X}$ düz metni $e_{\mathcal{K}}$ ile şifrelendiğinde elde edilen şifreli metnin şifresi $d_{\mathcal{K}}$ ile çözülerek yine $\mathcal{X}$ düz metni elde edilir.

Bazı temel şifreleme metodlarının isimlerini saymamız gerekirse; ilk olarak *Kaydırmalı Şifre* (Shift Cipher) akla gelir ki bu şifreleme metodu Roma imparatoru Julius Caesar tarafından da kullanılmıştır. Diğer metodlar ise *Yerine Koyma Şifresi* (Substitution Cipher), *Afin Şifresi* (Affine Cipher), *Vigenere Şifresi* (Vigenere Cipher), *Hill Şifresi* (The Hill Cipher), *Permütasyon Şifresi* (Permutation/ Transposition Cipher) ve *Dizi Şifreleyiciler* (Stream Ciphers) olarak sayılabilir.

Günümüz şifrelemesinde kullanılan metodlara ise bir sonraki bölümde değinilmiştir.

3.3. Kriptografi Sistemleri

Kriptografik sistemler genel olarak iki ana başlık altında incelenebilir; *Simetrik Anahtarlı Şifreleme Sistemleri* (Symmetric Key Cryptography Systems) ve *Açık Anahtarlı Şifreleme Sistemleri* (Public Key Cryptography Systems).

Simetrik Anahtarlı Şifreleme sistemlerinde haberleşen unsurlar öncelikli olarak gizli ve güvenilir bir anahtar üzerinde anlaşırlar. Ardından çeşitli simetrik anahtarlamalı şifreleme şemalarından uygun olanı kullanarak haberleşmeyi sağlarlar. Şekil 3.2'de Simetrik Anahtarlı Şifreleme ile ilgili şema görülmektedir.

Şekil 3.2 Simetrik anahtarlı şifreleme

Kısa bir örnekle bu şifreleme sistemine değinecek olursak; *A* ve *B* arasında paylaşılan gizli anahtar *k* ise, *A* *k* anahtarını ve $\mathcal{ENC}$ şifreleme fonksiyonunu kullanarak *m* düzmetnini şifreler ve elde ettiği *c* şifrelimetnini *B*'ye iletir.

$$c = \mathcal{ENC}_k(m)$$

c şifrelimetnini alan *B*, aynı *k* anahtarını ve $\mathcal{DEC}$ deşifreleme fonksiyonunu kullanarak *c* şifrelimetnini deşifreler ve *m* düzmetnini elde eder.

$$m = \mathcal{DEC}_k(c)$$

Şayet bilgi bütünlüğü ve bilgi kaynağının doğrulanması gerekiyorsa çeşitli algoritmalar kullanılarak bunların sağlanıp saplanmadığı kontrol edilebilir. (Örneğin; *Message Authentication Code* – MAC – Algoritması)

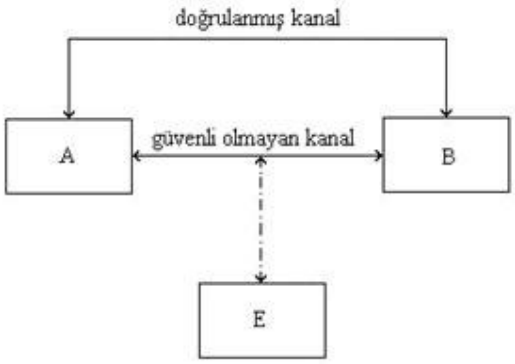
Simetrik Anahtarlamalı Şifreleme sistemlerinde *anahtar dağıtımı* (key distribution), *anahtar yönetimi* (key management) ve *inkar edememe* (non-repudiation) gibi konularda problemler yaşanmaktadır. Anahtar dağıtımının problemsiz gerçekleşmesi için gizli ve doğrulanmış bir kanal olması şartı bulunmaktadır. Birden fazla kullanıcının olduğu bir ağda, her bir kullanıcının farklı anahtar kullanması gerekliliği de anahtar yönetiminde yaşanması muhtemel sorunlardan biridir.

Açık Anahtarlı Şifreleme sistemleri, Simetrik Anahtarlı Şifreleme sistemlerinde ortaya çıkan güçlüklerin giderilmesi amacıyla icat edilmiş ve günümüzde en çok kullanılan sistem haline gelmiştir.

4. AÇIK ANAHTARLI ŞİFRELEME

4.1. Açık Anahtarlı Şifrelemenin Temelleri

Açık Anahtarlı Şifreleme (Public Key Cryptography – PKC) sisteminin tasarımı 1975 yılında Diffie, Hellman ve Merkle tarafından yapılmıştır. Bu sistemde haberleşen unsurlar için doğrulanmış bir kanal şarttır, ancak Simetrik Anahtarlı Şifreleme sistemlerinin tersine bu kanalın gizli olması şartı yoktur. Açık Anahtarlı Şifreleme sistemleri ile ilgili şema da Şekil 4.1'de görülebilir.



Şekil 4.1 Açık anahtarlı şifreleme

Bu sistemde her bir unsur (*e*, *d*) olmak üzere kendisi için bir anahtar çifti seçer. Bu anahtar çiftinde *e açık anahtar* ve *d gizli anahtar* olarak adlandırılır. Bu anahtarların özelliği, sadece açık anahtara sahip olarak gizli anahtarın çözülmesinin zor olmasıdır. PKC sistemlerinin çalışma prensibine göz atacak olursak; *A*, *B*'ye *m* düzmetnini göndermek istediği takdirde *B*'nin açık anahtarının kopyası olan e_B'yi ve açık anahtarlı şifreleme fonksiyonu $\mathcal{ENC}$'i kullanarak *m* metnini şifreleyerek *c* şifreli metnini elde eder.

$$c = \mathcal{ENC}_{e_B}(m)$$

B, c şifrelimetnini aldıktan sonra deşifreleme fonksiyonu $\mathcal{DEC}$'i ve kendi gizli anahtarı olan d_B'yi kullanarak *c'*yi deşifre eder ve *m* düzmetnini elde eder.

$m = DEC_{d_B}(c)$

Şifreleme ve deşifreleme işlemleri bu şekilde yapılarak *gizlilik* (confidentially) ilkesi de sağlanmış olur.

Şayet bu tip şifreleme işlemlerinde dijital imza sistemleri de kullanılacak olursa, işlemlerin sonucunun unsurlardan herhangi birisi tarafından *inkar edilmesinin* de önüne geçilmiş olur(non-repudiation). Bunun yanısıra *bilgi kaynağı doğrulaması* (data origin authentication) ve *bilgi bütünlüğü doğrulaması* da (data integrity) yapılmış olur.

Dijital İmza Sistemleri (digital signature schemes) konusuna kısaca değinilecek olursa; *A* kişisi *SIGN imza oluşturma algoritmasını* (signature generation algorithm), *m* düzmetnini ve kendine ait gizli anahtar d_A'yı kullanarak *s* imza mesajını oluşturabilir;

$s = SIGN_{d_A}(m)$

m ve *s*'i alan *B* halihazırda *A*'ya ait açık anahtar e_A'ya da sahiptir. Bunları kullanarak mesajın *A* tarafından gönderildiğini doğrulayabilir.

Bu sayede, Açık Anahtarlı Şifreleme sistemleri, anahtar dağıtımı, anahtar yönetimi ve inkar edilememe gibi Simetrik Anahtarlı Şifreleme sistemlerinde ortaya çıkan üç problemi de gidermiş durumdadır [2].

4.2. Açık Anahtarlı Şifreleme Algoritmaları

Açık Anahtarlı Şifreleme sistemleri, gizli anahtarın açık anahtar kullanılarak hesaplanmasının güçlüğüne dayanır. Buna dayanarak ve sayılar teorisinin de kullanılmasıyla ortaya çıkmış olan en önemli üç Açık Anahtarlı Şifreleme algoritması şunlardır;

• Tamsayıların çarpanlarına ayırma problemine (integer factorization problem) dayanan RSA Açık Anahtar Şifreleme ve İmza Sistemleri,
• Ayrık logaritma problemine (discrete logarithm problem) dayanan ElGamal Açık Anahtar Şifreleme ve İmza Sistemleri,
• Eliptik eğri ayrık logaritma problemine (elliptic curve discrete logarithm problem) dayanan Eliptik Eğri Açık Anahtar Şifreleme ve İmza Sistemleri.

Eliptik Eğri Açık Anahtar Şifreleme ve İmza Sistemleri tez kapsamında ilerleyen bölümlerde incelenecektir.

4.2.1. RSA şifreleme sistemleri

Bu sistemi isimlendirmek için, sistemi ortaya koymuş olan üç kişinin isimlerinin baş harfleri kullanılmıştır. Bu kişiler Rivest, Shamir ve Adleman'dır ve RSA, açık anahtarlı şifrelemenin ortaya çıkmasının hemen ardından, 1977 yılında uygulamaya sokulmuştur. Daha önceden de bahsedildiği üzere, RSA sisteminin temelinde tamsayı çarpanlara ayırma problemi bulunmaktadır.

Bu sistem $n \in \mathbb{Z}_n$'deki hesaplamalara dayanmaktadır. n sayısı iki farklı asal sayı olan p ve q'nun çarpımıyla elde edilmektedir. Ayrıca Euler Fonksiyonu $\phi(n)$'de hesaplamalarda kullanılmaktadır.

$$\phi(n) = (p-1)\cdot(q-1)$$

Sistemin detayları şu şekildedir; ($\mathcal{P}$, $\mathcal{C}$, $\mathcal{K}$, $\mathcal{E}$, $\mathcal{D}$)

i) p ve q asal sayılar olmak üzere, $n = p \cdot q$ hesaplanır.

ii) $\mathcal{P}, \mathcal{C} = \mathbb{Z}_n$ kabul edilir ve

$$\mathcal{K} = \{(n, p, q, a, b) : a \cdot b \equiv 1 \pmod{\phi(n)}\}$$

alınır.

iii) $\mathcal{K} = (n, p, q, a, b)$ ile

$e_{\mathcal{K}}(x) = x^b \pmod{n}$

ve

$d_{\mathcal{K}}(y) = y^a \pmod{n}$

tanımlanır. $(x,\ y \in \mathbb{Z}_n)$

Bu işlemler sonucunda elde edilen *n* ve *b*, *açık anahtarı* (public key) oluşturur. *p*, *q* ve *a* ise *gizli anahtar*lardır. (private key)

Şayet şifreleme ve deşifreleme işlemlerinin ters işlemler olduğunu göstermemiz gerekirse;

$a \cdot b \equiv 1 \pmod{\phi(n)}$

$t \geq 1$ alınırsa;

$a \cdot b \equiv t \cdot \phi(n) + 1$

$x \in \mathbb{Z}_n^*$ olduğundan

$$\begin{aligned}(x^b)^a &\equiv x^{t\cdot\phi(n)+1} \pmod{n} \\ &\equiv (x^{\phi(n)})^{\mathrm{t}} \cdot x \pmod{n} \\ &\equiv 1^t \cdot x \pmod{n} \\ &\equiv x \pmod{n}\end{aligned}$$

elde edilir.

4.1 Örnek

$p = 101$ ve $q = 113$ alınarak açık ve gizli anahtarlar hesaplanacak olursa;

$n = p \cdot q = 101 \cdot 113 = 11413$

$\phi(n) = (p-1) \cdot (q-1) = (101-1) \cdot (113-1) = 11200$

$11200 = 2^6 \cdot 5^2 \cdot 7$

$(\phi(n),\ b) = 1$ şartını sağlayan b sayısı ilgili algoritmalar kullanılarak hesaplanacak olursa;

$e \equiv 3533 \pmod{11200}$

$d^{-1} \equiv 6597 \pmod{11200}$

bulunur.

Elde edilen $e = 3533$ sayısı açık anahtar olarak kullanılabilir. $d = 6597$ sayısı ise gizli anahtardır.

Algoritma. RSA anahtar çifti oluşturma

Girdi. Güvenlik parametresi, *l.* (*l*, rasgele pozitif çift tamsayı)

Çıktı. RSA açık anahtarı (*n*, *e*) ve gizli anahtar *d.*

i) $l/2$ şartını sağlayan, eşit uzunlukta rasgele p ve q asal sayıları belirlenir.

ii) $n = p \cdot q$ ve $\phi = (p-1) \cdot (q-1)$ hesaplanır.

iii) $1 < e < \phi$ ve $(e,\ \phi) = 1$ şartlarını sağlayan keyfi e sabiti belirlenir,

iv) $1 < d < \phi$ ve $e \cdot d \equiv 1 \pmod{\phi}$ şartlarını sağlayan d sayısı hesaplanır,

v) (n, e, d) bulunmuş olur.

RSA şifreleme sistemi tüm m tamsayıları için;

$m^{e \cdot d} \equiv m \pmod{n}$

prensibine dayanmaktadır. Temel RSA şifreleme ve deşifreleme algoritmaları ise şu şekildedir.

Algoritma. Temel RSA şifrelemesi
Girdi. RSA açık anahtarı (n, e), düzmetin $m \in [0,\ n-1]$.
Çıktı. Şifrelimetin c.

i) $c \equiv m^e \pmod{n}$ hesaplanır.
ii) c bulunmuş olur.

Algoritma. Temel RSA deşifrelemesi
Girdi. RSA açık anahtarı (n, e), RSA gizli anahtarı d, şifrelimetin c.
Çıktı. Düzmetin m.

i) $m \equiv c^d \pmod{n}$ hesaplanır.
ii) m bulunmuş olur.

Tüm bu algoritmaların yanısıra matematiğin diğer birçok temel teoremleri ve algoritmaları da kullanılmaktadır. En büyük ortak bölen ve Euclid Algoritması gibi birçok konu bu işlemlerin yapılması sırasında direkt olarak ya da yardımcı olması açısından kullanılmaktadır.

5. ELİPTİK EĞRİLER

5.1. Gerçel Sayılar Üzerinde Tanımlı Eliptik Eğriler

Gerçel sayılar üzerinde *E* eliptik eğrisini tanımlayabilmek için, seçilmiş olan (*x*, *y*) noktasının aşağıda bulunan eliptik eğri denklemini sağlaması gerekmektedir;

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

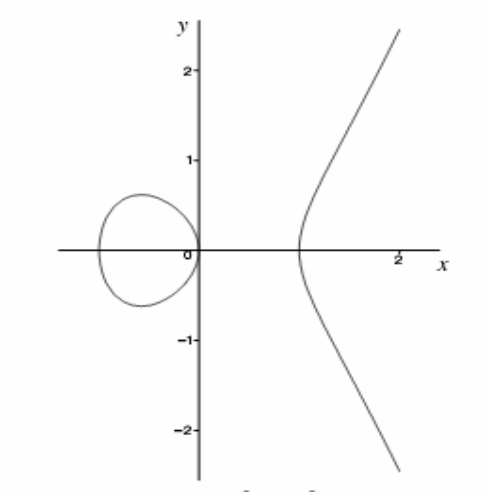
Burada *x*, *y*, *a* ve $b \in \mathbb{R}$ olmalıdır. Yukarıdaki denklem *Weierstrass Denklemi* olarak adlandırılır. Farklı olarak seçilen *a* ve *b* değerleri farklı bir eliptik eğrinin oluşturulmasını sağlar. Örneğin,

$a = -1, \mathrm{b} = 0$

değerleri seçilerek oluşturulan eliptik eğri denklemi şu şekilde olacaktır:

$$y^2 = x^3 - x$$

Elde edilen eliptik eğrinin grafiği ise Şekil 5.1'de görülebilir.

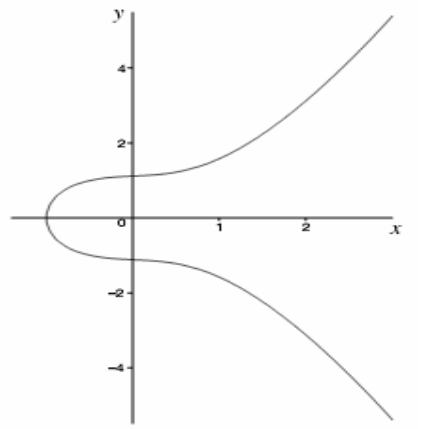


Şekil 5.1 $y^2 = x^3 - x$ eğri grafiği

Şayet $x^3 + a \cdot x + b$ katlı köke sahip değilse, ya da bir başka deyişle;

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

ise, $y^2 = x^3 + a \cdot x + b$ eliptik eğrisi bir *grup* oluşturur. Gerçel sayılar üzerindeki bir eliptik eğri grubu $\mathcal{O}$ olarak adlandırılan bir *sonsuzluk noktası*'na sahiptir. (point at infinity) Diğer bir eliptik eğri örneği ise Şekil 5.2'de görülebilir.



Şekil 5.2 $y^2 = x^3 + \frac{1}{4}x + \frac{5}{4}$ eğri grafiği

5.1.1. Eliptik eğri nokta toplama: geometrik yaklaşım

Eliptik eğri grupları toplamsal gruplardır. Bundan dolayı bu grupların temel işlemi toplamadır. Eliptik eğri üzerindeki iki noktanın toplanması geometrik olarak tanımlanmıştır.

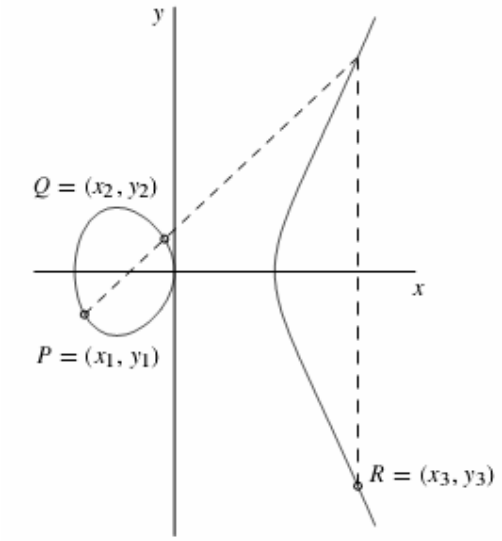
$P(x_P, y_P)$ noktasının, bulunduğu eliptik eğriye göre negatifini elde etmek için bu noktanın x eksenine göre simetriği alınır ve $-P(x_P, -y_P)$ olarak elde edilir. Eğri üzerinde bulunan tüm noktaların negatifleri de yine eğri üzerindedir.

i) *P ve Q olmak üzere iki farklı noktanın toplanması*: P ve Q noktalarının eğri üzerinde farklı noktaları temsil ettiğini ve $P \neq -Q$ olduğunu kabul edersek; P ve Q

noktalarını birleştiren çizgi, eğriyi üçüncü bir noktada keser. Çizginin eğriyi kestiği bu nokta $-R$ olarak adlandırılır. Şayet $-R$ noktasının x eksenine göre simetriği alınırsa R noktası elde edilir. Eliptik eğri nokta toplama işleminin kuralı şu şekildedir;

$$P + Q = R$$

Bu işlem Şekil 5.3'de görülebilir.



Şekil 5.3 $P + Q = R$ toplama işlemi (geometrik yaklaşım)

5.1 Örnek

$P = (-2.35, -1.86)$

$Q = (-0.1, 0.836)$

$-R = (3.89, 5.62)$

$R = (3.89, -5.62)$

$P + Q = R = (3.89, -5.62)$

ii) *P ve –P noktalarının toplanması*: *P* ve *–P* noktalarını birleştiren çizgi, eğriyi üçüncü bir noktada kesmez. Buna göre *P* ve *–P* noktaları bir önceki maddede yapılan toplama işlemi gibi toplanamaz. Bu sayede eliptik eğri, sonsuzluk noktası $\mathcal{O}$'yu içerir. Eşitlik olarak yazmak gerekirse;

$$P + (-P) = \mathcal{O}$$

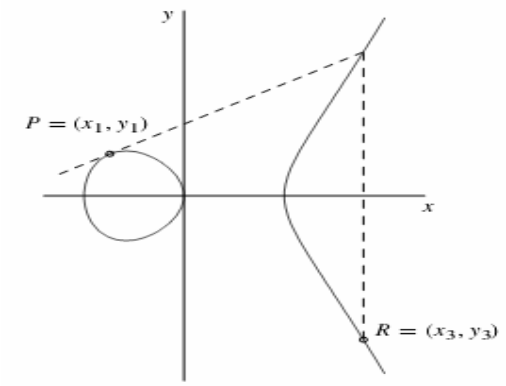
elde edilir. Bu eşitliğin sonucu olarak;

$$P + \mathcal{O} = P$$

yazılabilir. $\mathcal{O}$, eliptik eğri grubunun toplamsal birimi (sıfırı) olarak adlandırılır. Tüm eliptik eğriler için toplamsal özdeşlik vardır [1].

P noktasının çiftlenmesi: Bir noktayı kendisi ile toplamak için, bu noktadan eliptik eğriye teğet çizildiğinde eğer $y_P \neq 0$ ise bu teğet eliptik eğriyi *–R* olarak adlandırılan ikinci bir noktada keser. Şayet *–R* noktasının *x* eksenine göre simetriği alınırsa *R* noktası elde edilir. Bu işlem Şekil 5.4'de görülebilir. Eşitlik olarak yazmak gerekirse;

$$P + P = 2 \cdot P = R$$



Şekil 5.4 *P* noktasının çiftlenmesi (geometrik yaklaşım)

iii) $y_P = 0$ *ise* P *noktasının çiftlenmesi*: $y_P = 0$ olan bir P noktasının kendisi ile toplanması durumunda eliptik eğriye çizilen teğet, dikey bir çizgi olacaktır ve teğet olarak çizilen bu çizgi eğriyi başka herhangi bir noktada kesmeyecektir. Eşitlik olarak yazmak gerekirse;

$$P + P = 2 \cdot P = \mathcal{O}$$

Bu durumda $3 \cdot P = 2 \cdot P + P$ bulunmak istenirse; $P + \mathcal{O} = P$'ye dayanarak $3 \cdot P = P$ bulunur. Örneğin;

$$\begin{aligned} P + P &= 2 \cdot P = \mathcal{O} \\ 2 \cdot P + P &= 3 \cdot P = \mathcal{O} + P = P \\ 3 \cdot P + P &= 4 \cdot P = P + P = \mathcal{O} \\ 4 \cdot P + P &= 5 \cdot P = \mathcal{O} + P = P \end{aligned}$$

5.1.2. Eliptik eğri nokta toplama: cebirsel yaklaşım

Eliptik eğri nokta toplama ile ilgili bir önceki bölümde yer alan geometrik yaklaşım, eliptik eğri aritmetiğinin ortaya koyulması açısından oldukça faydalıdır. Ancak aritmetik hesaplamalar için geometrik yaklaşım pratik bir çözüm olmadığından, cebirsel formüller oluşturularak bu hesaplamalar daha verimli bir şekilde yapılmaktadır.

i) P *ve* Q *olmak üzere iki farklı noktanın toplanması*: Şayet $P = (x_P, y_P)$, $Q = (x_Q, y_Q)$ ve $P \neq -Q$ olarak kabul edilecek olursa;

$$P + Q = R$$

için

$$s = \frac{y_P - y_Q}{x_P - x_Q}$$

$$x_R = s^2 - x_P - x_Q$$

ve

$$y_R = -y_P + s \cdot (x_P - x_R)$$

olarak bulunabilir. s, P noktası ile Q noktasını birleştiren çizginin eğimidir.

ii) *P noktasının çiftlenmesi*: $y_P \neq 0$ ise;

$$P + P = 2 \cdot P = R$$

için

$$s = \frac{3 \cdot x_P^2 + a}{2 \cdot y_P}$$

$$x_R = s^2 - 2 \cdot x_P$$

ve

$$y_R = -y_P + s \cdot (x_P - x_R)$$

olarak bulunabilir. *a*, eliptik eğri oluşturulurken seçilen sabit ve *s*, *P* noktasından eliptik eğriye çizilen teğetin eğimidir.

Konu ile olarak aşağıdaki örnekler verilebilir:

5.2 Örnek

$y^2 = x^3 - 17 \cdot x + 16$ eliptik eğrisini göz önüne alalım.

i) Eğri, $4 \cdot a^3 + 27 \cdot b^2 \neq 0$ şartını sağlamaktaysa, $\mathbb{R}$ üzerinde bir grup tanımlamaktadır.

$y^2 = x^3 - 17 \cdot x + 16$

a = –17

b = 16

$4 \cdot a^3 + 27 \cdot b^2 = 4 \cdot (-17)^3 + 27 \cdot 16^2 = 6912 \neq 0$

ii) $P = (0, -4)$ ve $Q = (1, 0)$ alınarak $P + Q = R$ hesaplanacak olursa;

$P + Q = R$

için

$$
\left.
\begin{aligned}
& s = \frac{y_P - y_Q}{x_P - x_Q} \\
& s = \frac{-4 - 0}{0 - 1} = 4 \\
& x_R = s^2 - x_P - x_Q \\
& x_R = 4^2 - 0 - 1 \\
& x_R = 15 \\
& y_R = -y_P + s \cdot (x_P - x_R) \\
& y_R = -(-4) + 4 \cdot (0 - 15) \\
& y_R = -56
\end{aligned}
\right\} P + Q = R = (15, -56)
$$

iii) $P = (4, 3.464)$ alınarak $P + P = 2 \cdot P = R$ hesaplanacak olursa;

$P + P = 2 \cdot P = R$

için

$$
\left.
\begin{aligned}
s &= \frac{3 \cdot x_P^2 + a}{2 \cdot y_P} \\
s &= \frac{3 \cdot 4^2 - 17}{2 \cdot 3.464} = 4.48 \\
x_R &= s^2 - 2 \cdot x_P \\
x_R &= 4.48^2 - 2 \cdot 4 \\
x_R &= 12.02 \\
y_R &= -y_P + s \cdot (x_P - x_R) \\
y_R &= -3.464 + 4.48 \cdot (4 - 12.02) \\
y_R &= -39.39
\end{aligned}
\right\} P + P = 2 \cdot P = R = (12.02, -39.39)
$$

Eliptik eğrilerin sonlu sayıda noktadan oluşan bir grup olması, kriptografik uygulamalar için önemli bir özelliktir. Kriptografik uygulamalar yapılırken, sonuçlarda herhangi bir sapmanın ya da yuvarlama işleminin olmaması gerekmektedir ve eliptik eğri grupları kullanılarak yapılan işlemler neticesinde elde edilen sonuçlar yine bu grup içerisinden elemanlar olacağından sonuçlarda herhangi bir hata olma olasılığı bulunmamaktadır.

5.2. F_P Üzerinde Tanımlı Eliptik Eğriler

Gerçel sayılar üzerinde yapılan hesaplamalar yavaş olmaktadır ve yuvarlamadan dolayı hatalı sonuçlar verebilmektedir. Bunun tersine, kriptografik uygulamalar hızlı ve hassas hesaplamalar gerektirmektedir. Bu sebepten dolayı, eliptik eğriler pratikte genellikle F_P ve F_{2^m} üzerinde uygulanırlar. F_P, 0 ve p – 1 arasındaki sayılardan oluşur ve hesaplamalar p ile bölümden elde edilen kalana göre yapılır. Örneğin F_{23}, 0, 1, ..., 21, 22 sayılarından oluşur. Bu cisimde (field) yapılan herhangi bir işlemin sonucu 0 ile 22 arasındaki bir tamsayı olur.

F_P'de bir E eliptik eğrisi tanımlamak için seçilecek a ve b sabitleri F_P içerisinden seçilmelidir. (x, y) noktaları, eliptik eğri üzerinde mod p'ye göre hesaplanmış noktalardır. (x ve y, F_P'de tanımlı tamsayılardır.)

Örneğin,

$$y^2 \pmod p \equiv x^3 + a \cdot x + b \pmod p$$

eliptik eğrisi ve *a* ve *b* sabitleri F_P'de tanımlıdır.

Şayet $x^3 + a \cdot x + b$ katlı köke sahip değilse $(4 \cdot a^3 + 27 \cdot b^2 \neq 0 \pmod p)$, bu eliptik eğri bir grup oluşturur. Bu grup, F_P üzerinde tanımlı eliptik eğride bulunan noktalar ve sonsuzluk noktası $\mathcal{O}$'yu içerir. Bu eliptik eğri üzerinde sınırlı sayıda nokta bulunmaktadır.

5.2.1. F_P üzerinde tanımlı bir eliptik eğri

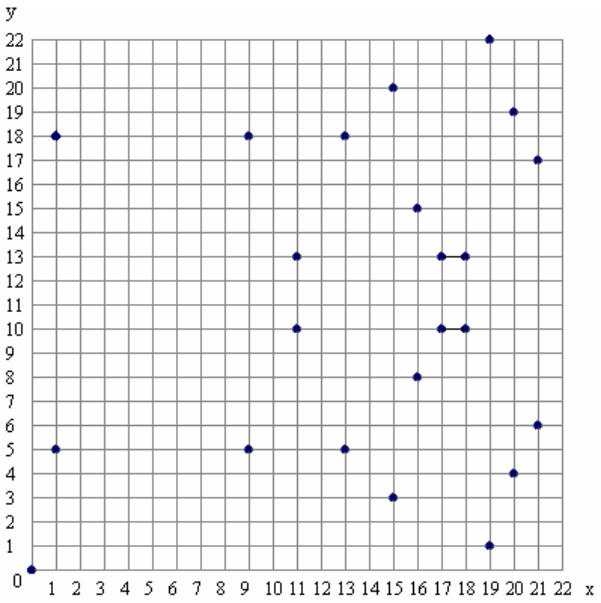
$a = 1$ ve $b = 0$ olmak üzere, F_{23}'te tanımlı eliptik eğri örneği göz önüne alınacak olursa, (9, 5) noktasının aşağıdaki eşitliği sağladığı görülmektedir.

$$\begin{aligned} y^2 \pmod p &= x^3 + x \pmod p \\ 5^2 \pmod{23} &= 9^3 + 9 \pmod{23} \\ 25 \pmod{23} &= 738 \pmod{23} \\ 2 &= 2 \end{aligned}$$

Bu denklemi sağlayan 23 nokta şunlardır;

(0, 0), (1, 5), (1, 18), (9, 5), (9, 18), (11, 10), (11, 13), (13, 5), (13, 18),
(15, 3), (15, 20), (16, 8), (16, 15), (17,10), (17, 13), (18, 10), (18, 13),
(19, 1), (19, 22), (20, 4), (20, 19), (21, 6), (21, 17)

Şekil 5.5'de bu noktaların dağılımı görülmektedir.



Şekil 5.5 F_{23}'te tanımlı $y^2 = x^3 + x$ grafiği

Görüleceği gibi her x değerine karşılık iki nokta bulunmaktadır. Grafik rasgele görünmekte ise de y = 11.5'e göre bir simetri mevcuttur. Şayet gerçel sayılar üzerinde tanımlı eliptik eğrileri düşünecek olursak, bu eğriler üzerinde tanımlı olan her noktanın x eksenine göre simetriği alınarak elde edilen bir negatifi mevcuttur. F_{23}'te tanımlı eliptik eğri üzerinde bulunan noktalar için simetrik noktalar (negatifleri) y değerlerinin mod 23'ü alınarak elde edilir.

$$-P = (x_P,\ (-y_P \pmod{23}))$$

Yukarıda bahsedilen tüm kurallar gerçel sayılar üzerinde tanımlı eliptik eğriler ile benzerlikler göstermektedir. Aradaki fark ise bu eğriler üzerindeki işlemlerin mod p'de yapılmasıdır.

5.2.2. F_P üzerinde tanımlı eliptik eğrilerde aritmetik

F_P üzerinde tanımlı eliptik eğri grupları ile gerçel sayılar üzerinde tanımlı eliptik eğri grupları arasında büyük farklılıklar bulunmaktadır. F_P üzerinde tanımlı eliptik eğri gruplarında sonlu sayıda nokta bulunmaktadır ve bu özellik kriptografik uygulamalar için oldukça elverişlidir. Bu eğriler sayıları az olan farklı noktalar içermektedir, fakat bu noktaların birleştirilmesi, bir eğri görünümüne kavuşturulması ve sonuç olarak geometrik ilişkiler çok açık değildir. Gerçel sayılar üzerinde tanımlı eliptik eğriler için geçerli olan geometrik yaklaşımlar F_P üzerinde tanımlı eliptik eğriler için geçerli değildir. Ancak gerçel sayılar üzerinde tanımlı eliptik eğriler üzerinde işlem yapılırken ortaya çıkması muhtemel yuvarlama hataları F_P üzerinde tanımlı eliptik eğriler için söz konusu değildir. Daha önce de değinildiği gibi bu özellik kriptografik uygulamalar için oldukça önemlidir [1].

i) *P ve Q olmak üzere iki farklı noktanın toplanması*: $P = (x_P, y_P)$ noktasının negatifi $-P = (x_P,\ (-y_P \pmod{\text{p}}))$'dir. $P \neq -Q$ olmak üzere iki farklı nokta ise;

$$P + Q = R$$

için

$$s = \frac{y_P - y_Q}{x_P - x_Q} \pmod{p}$$

$$x_R = s^2 - x_P - x_Q \pmod{p}$$

ve

$$y_R = -y_P + s \cdot (x_P - x_R) \pmod{p}$$

olarak bulunur. s, P noktası ile Q noktasını birleştiren çizginin eğimidir.

ii) *P noktasının çiftlenmesi*: $y_P \neq 0$ ise;

$$P + P = 2 \cdot P = R$$

için

$$s = \frac{3 \cdot x_P^2 + a}{2 \cdot y_P} \pmod{p}$$

$$x_R = s^2 - 2 \cdot x_P \pmod{p}$$

ve

$$y_R = -y_P + s \cdot (x_P - x_R) \pmod{p}$$

olarak bulunur. *a*, eliptik eğri oluşturulurken seçilen sabit ve *s*, *P* noktasından eliptik eğriye çizilen teğetin eğimidir.

5.3 Örnek

F_{17} üzerinde tanımlı $y^2 = x^3 + x + 7$ eliptik eğrisini alacak olursak;

i) Eğri, $4 \cdot a^3 + 27 \cdot b^2 \neq 0$ şartını sağlamaktaysa, F_{17} üzerinde bir grup tanımlamaktadır.

$$y^2 = x^3 + x + 7$$

a = 1

b = 7

$$4 \cdot a^3 + 27 \cdot b^2 = 4 \cdot 1^3 + 27 \cdot 7^2 = 1327 \pmod{17} \neq 0$$

ii) $P = (2, 0)$ ve $Q = (1, 3)$ alınarak $P + Q = R$ hesaplanacak olursa;

$$P + Q = R$$

için

$$
\left.
\begin{aligned}
& s = \frac{y_P - y_Q}{x_P - x_Q} \pmod{17} \\
& s = \frac{0-3}{2-1} \equiv 14 \pmod{17} \\
& x_R = s^2 - x_P - x_Q \pmod{17} \\
& x_R = 14^2 - 2 - 1 \pmod{17} \\
& x_R \equiv 6 \pmod{17} \\
& y_R = -y_P + s \cdot (x_P - x_R) \pmod{17} \\
& y_R = -0 + 14 \cdot (2-6) \pmod{17} \\
& y_R \equiv 12 \pmod{17}
\end{aligned}
\right\} P + Q = R = (15, -56) \pmod{17}
$$

iii) $P = (1, 3)$ alınarak $P + P = 2 \cdot P = R$ hesaplanacak olursa;

$$P + P = 2 \cdot P = R$$

için

$$
\left.
\begin{aligned}
& s = \frac{3 \cdot x_P^2 + a}{2 \cdot y_P} \pmod{17} \\
& s = \frac{3 \cdot 1^2 + 1}{2 \cdot 3} \equiv 12 \pmod{17} \\
& x_R = s^2 - 2 \cdot x_P \pmod{17} \\
& x_R = 12^2 - 2 \cdot 1 \pmod{17} \\
& x_R \equiv 6 \pmod{17} \\
& y_R = -y_P + s \cdot (x_P - x_R) \pmod{17} \\
& y_R = -3 + 12 \cdot (1-6) \pmod{17} \\
& y_R \equiv 5 \pmod{17}
\end{aligned}
\right\} P + P = 2 \cdot P = R = (6, 5) \pmod{17}
$$

6. ELİPTİK EĞRİ GRUPLARI VE AYRIK LOGARİTMA PROBLEMİ

Tüm kriptografik sistemlerin temelinde, pratikte hesaplama yolu ile çözümü oldukça zor olan matematiksel problemler yatmaktadır. Ayrık logaritma problemi, Eliptik Eğri Şifreleme sistemlerinin de dahil olduğu birçok kriptografik sistemin temelini oluşturmaktadır. Eliptik Eğri Şifreleme sistemleri *Eliptik Eğri Ayrık Logaritma Problemi*'ne dayanmaktadır. (Elliptic Curve Discrete Logarithm Problem – ECDLP) Daha önceki bölümlerde de değinildiği gibi, eliptik eğri grupları üzerinde tanımlı çeşitli aritmetik işlemler bulunmaktadır. Bu işlemler *nokta toplama* ve *nokta çiftlemedir*.

Eliptik eğri grubu üzerinde seçilen bir P noktasının kendisi ile toplanması sonucunda (nokta çiftleme) $2 \cdot P$ elde edilir. Bu işlem yapıldıktan sonra, yine aynı P noktası $2 \cdot P$ ile toplandığında $3 \cdot P$ elde edilir. Bu işlemin ardarda yapılması sonucu $n \cdot P$ noktası elde edilir. Yapılan işlem de *Skaler Çarpma* (Scalar Multiplication) olarak adlandırılır. Eliptik Eğri Ayrık Logaritma Problemi, skaler çarpma sonucu elde edilen noktaların takibinin zorluğuna dayanmaktadır [1].

6.1. Skaler Çarpma

Bir eliptik eğri grubu tanımlanırken toplamsal notasyon kullanılır. Bu konunun biraz daha detaylarına inilmesi durumunda ise çarpımsal notasyon elde edilebilir. Çarpımsal notasyon, toplamsal notasyon kullanılarak tanımlanacak olursa; $k \cdot P$'nin hesaplanması, P noktasının k defa kendisi ile toplanmasıdır. Çarpımsal notasyon, P noktasının k defa kendisi ile çarpılması olarak da yazılabilir.

$$P \cdot P \cdot P \cdot P \cdot \ldots = k \cdot P$$

6.2. Ayrık Logaritma Problemi

Z_P^* çarpımsal grubu göz önüne alınarak ayrık logaritma problemi açıklanacak olursa; p asal sayısı, r ve q da verilen elemanlar olmak üzere,

$r = q \cdot k \pmod{p}$

eşitliğini sağlayan bir *k* değeri bulma işlemidir.

Şayet eliptik eğri grubu çarpımsal notasyon kullanılarak tanımlanacak olursa, bu durumda *P* ve *Q* noktaları için ortaya çıkan Eliptik Eğri Ayrık Logaritma Problemi,

$P \cdot k = Q$

eşitliğini sağlayan bir *k* değeri bulma işlemidir ve *Q noktasının P tabanında ayrık logaritması* olarak adlandırılır.

6.1 Örnek

F_{23}'te tanımlanmış olan $y^2 = x^3 + 9 \cdot x + 17$ eliptik eğrisi üzerinde bulunan *Q* = (4, 5) noktasının *P* = (16, 5) tabanına göre ayrık logaritması olan *k* hesaplanacak olursa; çözümün elde edilmesi için yapılacak işlem basitçe *P* noktasının katlarının *Q* noktası elde edilene kadar toplanmasıdır. *P* noktasının birkaç katı şu şekilde olacaktır;

$P = (16,\ 5),\ 2 \cdot P = (20,\ 20),\ 3 \cdot P = (14,\ 14),\ 4 \cdot P = (19,\ 20),\ 5 \cdot P = (13,\ 10),$
$6 \cdot P = (7,\ 3),\ 7 \cdot P = (8,\ 7),\ 8 \cdot P = (12,\ 17),\ 9 \cdot P = (4,\ 5)$

$9 \cdot P = (4,\ 5) = Q$ olduğundan, *Q*'nun *P* tabanına göre hesaplanan ayrık logaritması *k* = 9 olacaktır.

Gerçek bir kriptografik uygulamada kullanılacak olan *k* ise, hesaplaması pratik olarak zor olacak büyüklükte bir değer olarak seçilir.

Eliptik eğriler ile ilgili aritmetik işlemlere değindikten sonra aşağıda yer alan bazı kavramlara da değinmek yerinde olacaktır.

6.2.1. Karesel kalanlar

6.1 Tanım (Karesel Kalanlar)

p'nin asal bir sayı olduğunu kabul edecek olursak, şayet $a \neq 0 \pmod{p}$ sağlanıyorsa ve $y^2 \equiv a \pmod{p}$ kongrüansının $y \in Z_p$ şeklinde bir çözümü varsa *a*, mod *p*'de *karesel kalan* (quadratic residue) olarak tanımlanır. Şayet bu şekilde bir çözüm mevcut değilse *a* bir karesel kalan değildir [8].

6.2 Örnek

$\mathbb{Z}_{11}$'deki bazı sayıların karesi şu şekildedir;

$1^2 = 1,\ 2^2 = 4,\ 3^2 = 9,\ 4^2 = 5,\ 5^2 = 3,\ 6^2 = 3,\ 7^2 = 5,\ 8^2 = 9,\ 9^2 = 4,\ 10^2 = 1$

Mod 11'e göre karesel kalanlar, 1, 3, 4, 5, 9 ve karesel olmayan kalanlar 2, 6, 7, 8 ve 10 şeklinde listelenebilir.

p asal bir sayı ve *a* karesel kalan olarak kabul edilecek olursa, $y^2 \equiv a \pmod{p}$ denkleminin bir çözümü $y \in Z_p^*$ mevcuttur. Benzer olarak $(-y)^2 \equiv a \pmod{p}$ yazılabilir ancak *p* asal olduğundan $y \neq -y \pmod{p}$ şeklindedir. Karesel kongrüans olan $x^2 - a \equiv 0 \pmod{p}$,

$$(x - y) \cdot (x + y) \equiv 0 \pmod{p}$$

şeklinde faktörize edilebilir ve şu şekilde de yazılabilir:

$$p \mid (x - y) \cdot (x + y).$$

p asal olduğundan,

$$p \mid (x - y),\ p \mid (x + y)$$

olduğu görülebilir. Buradan da,

$$x \equiv \pm y \pmod{p}$$

sonucuna ulaşılabilir. Yukarıda yapılan işlemlerden elde edilen sonuca göre mod p'ye göre iki çözüm bulunmaktadır. Bu iki çözüm yine mod p'ye göre birbirinin negatifleridir.

6.2.2. Legendre sembolü

a tamsayısının mod p'ye göre karesel kalan olup olmadığı probleminin çözümü için *Euler Kriteri* (Euler's Criterion) kullanılabilir.

6.1 Teorem. (Euler Kriteri)

p'nin asal bir sayı olduğu kabul edilirse;

$$a^{(p-1)/2} \equiv 1 \pmod{p}$$

şartını sağlayan a karesel kalandır. Aşağıda yapılan işlemler sonucunda Euler Kriteri'nin doğruluğu görülebilir [8].

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

ve

$$a^{p-1} \equiv 1 \pmod{p} \quad (a \neq 0 \pmod{p})$$

$$\begin{aligned} a^{(p-1)/2} &\equiv (y^2)^{(p-1)/2} \pmod{p} \\ &\equiv y^{p-1} \pmod{p} \\ &\equiv 1 \pmod{p} \end{aligned}$$

$a^{(p-1)/2} \equiv 1 \pmod{p}$ olduğu kabul edilirse ve b mod p'ye göre primitif eleman ise; herhangi bir i tamsayısı için $a \equiv b^i \pmod{p}$ olur. Sonuç olarak;

$$a^{(p-1)/2} \equiv (b^i)^{(p\text{-}1)/2} \pmod{p}$$
$$\equiv b^{i\cdot(p\text{-}1)/2} \pmod{p}$$

elde edilir.

b, $p-1$ şeklinde bir üsse sahiptir ve i çift sayı ve a'nın çift katlı kökleri $\pm b^{i/2} \bmod p$ olduğundan $p-1$, $i\cdot(p-1)/2$'yi böler.

6.2 Tanım (Legendre Sembolü)

p bir asal sayı ise, herhangi bir a tamsayısı için *Legendre Sembolü* (Legendre Symbol) $\left(\frac{a}{p}\right)$ şu şekildedir;

$$\left(\frac{a}{p}\right) = \begin{cases} 0, & a \equiv 0 \pmod{p} \\ 1, & \text{şayet } a \bmod p\text{'ye göre karesel kalan ise} \\ -1, & \text{şayet } a \bmod p\text{'ye göre karesel kalan değilse} \end{cases}$$

Eliptik eğriler ile yapılan işlemler kapsamında daha hızlı sonuç elde edebilmek için Euler Kriteri ve Legendre Sembolü'nün yanısıra Z_n'de çift katlı kök hesaplamasını da verimli bir şekilde yapmak gereklidir. Bu işlemler aşağıdaki algoritmalar yardımı ile yapılabilir.

Algoritma. Çift Katlı Kök Hesabı
Girdi. $r^2 \equiv a \pmod{p}$ $(1 \le a \le p-1)$
Çıktı. a'nın çift katlı iki kökü.

1. Şayet $p \equiv 3 \pmod{4}$ ise;
a. $r \equiv a^{(p+1)/4} \pmod{p}$ hesaplanır.

b. (r, –r) sonuç olarak bulunur.

2. Şayet $p \equiv 5 \pmod 8$ ise;

a. $d \equiv a^{(p-1)/4} \pmod p$ hesaplanır.

b. Şayet $d \equiv 1 \pmod p$ ise

$r \equiv a^{(p+3)/8} \pmod p$ hesaplanır.

c.Şayet $d \equiv p-1 \pmod p$ ise

$r \equiv 2 \cdot a \cdot (4 \cdot a)^{(p-5)/8} \pmod p$

d. (r, –r) sonuç olarak bulunur.

3. Diğer tüm durumlar için;

a. Legendre Sembolü $\left(\frac{a}{p}\right) = -1$ ise a'nın çift katlı kökü yoktur. Sonlandırılır.

b. $1 \le b \le p-1$ ve $\left(\frac{b}{p}\right) = -1$ şartını sağlayan rasgele bir b sayısı seçilir.

c. p – 1 şekildeki notasyonda yazılır.

d. $p-1 = 2^s \cdot t$, (t tek sayı)

e. $a^{-1} \pmod p$ Euclid Algoritması ile hesaplanır.

f. $c \leftarrow b^t \pmod p$ ve $r \leftarrow a^{(t+1)/2} \pmod p$.

g. i, 1'den s – 1'e kadar

h. $d \equiv (r^2 \cdot a^{-1})^{2^{s-i-1}} \pmod p$ hesaplanır.

i. Şayet $d \equiv 1 \pmod p$ ise $r \leftarrow r \cdot c \pmod p$

j. $c \leftarrow c^2 \pmod p$

k. (r, –r) sonuç olarak bulunur.

Yukarıda değinilen konuları bir örnek yardımı ile açıklamak yerinde olacaktır.

6.3 Örnek

Z_{11}'de tanımlı E: $y^2 = x^3 + x + 6$ göz önüne alınarak E eliptik eğrisi üzerinde bulunan noktaların hesaplanması için yapılması gereken ilk işlem

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

eşitsizliğinin çözümünün bulunması olacaktır. Verilen herhangi bir x mod 11 için $z = x^3 + x + 6$ mod 11 çözümü olup olmadığının bulunması için yukarıda bahsedilen Euler Kriteri, Legendre Sembolü ve Çift Katlı Kök Algoritması uygulanabilir. Yapılacak uygulamalar sonucu elde edilecek değerler
Çizelge 6.1'de görülmektedir.

Çizelge 6.1 E: $y^2 = x^3 + x + 6$ üzerindeki çift katlı kökler

x	$x^3 + x + 6 \pmod{11}$	karesel kalan mı?	y
0	6	h	-
1	8	h	-
2	5	e	4, 7
3	3	e	5, 6
4	8	h	-
5	4	e	2, 9
6	8	h	-
7	4	e	2, 9
8	9	e	3, 8
9	7	h	-
10	4	e	2, 9

6.3. Eliptik Eğri Şifreleme Sistemleri

Eliptik eğrileri kullanan şifreleme sistemleri Açık Anahtarlı Şifreleme Sistemleri'dir. Sadece açık anahtarların bilinmesi ile şifre çözme işleminin yapılması pratikte

mümkün değildir. Bu tarz güvenliğin sağlanması için Açık Anahtarlı Şifreleme Sistemleri'nde ayrık logaritma problemi kullanılmaktadır.

Eliptik Eğriler kullanılarak yapılacak olan şifreleme işlemlerinde Eliptik Eğri Ayrık Logaritma Problemi temel olarak alınmaktadır. Temel alınan P noktasının skaler çarpma yöntemi ile elde edilen katları kullanılarak şifreleme yapılacak materyalin güvenliği sağlanmaya çalışılmaktadır. Bu noktada Eliptik Eğri Şifreleme metodunda kullanılan parametrelere göz atmak yerinde olacaktır.

E eliptik eğrisi, eğrinin tanımlı olduğu asal cisim p, P noktası, skaler çarpımlar sonucu elde edilen Q noktası *açık anahtar*lardır. Bu parametrelerin tek tek ya da tümünün birden bilinmesi şifrelemenin güvenliği hakkında herhangi bir risk unsuru oluşturmaz. Eliptik eğri şifrelemesindeki en önemli parametre, *gizli anahtar* olan m çarpanıdır. P noktasının ya da skaler çarpımlar sonucu elde edilen Q noktasının biliniyor olması m çarpanının bulunması için yeterli değildir. Ancak bu güvenliğin sağlanabilmesi için m değeri pratik olarak hesaplaması mümkün olmayacak şekilde seçilmelidir. Ayrık logaritma probleminin sağladığı güvenlik seviyesi de burada devreye girer [1].

Şifreleme işleminde kullanılacak olan algoritma ise şu şekilde olacaktır:

Algoritma. Eliptik Eğri Şifreleme Algoritması
Girdi. E eliptik eğrisi, P noktası (açık anahtar), m gizli anahtarı, x düzmetni, rasgele seçilen k tamsayısı
Çıktı. y şifrelimetni

i) İlk olarak Q noktası hesaplanır.

$$Q = m \cdot P$$

ii) Şifreleme işlemi için

$$e_K(x,\ k) = (k \cdot P,\ x + k \cdot Q)$$

uygulanır.

iii) $y_1 = k \cdot P$

ve

$y_2 = x + k \cdot Q = x + k \cdot m \cdot P$

şeklinde elde edilir.

iv) Şifrelimetin *y;*

$y = (y_1,\ y_2)$

şeklinde elde edilir.

Deşifreleme işleminde kullanılacak olan algoritma ise şu şekilde olacaktır.

Algoritma. Eliptik Eğri Deşifreleme Algoritması
Girdi. m gizli anahtarı, *y* şifrelimetni
Çıktı. x düzmetni

i) Deşifreleme işlemi için

$d_K(y_1,\ y_2) = y_2 - m \cdot y_1 = x$

uygulanır.
x düzmetni elde edilmiş olur.

Eliptik eğriler kullanılarak yapılan şifreleme metoduna basit bir örnek verilerek değinilmesi uygun olacaktır.

6.4 Örnek

Z_{11}'de tanımlı E: $y^2 = x^3 + x + 6$ eğrisi ve $m = 7$ gizli anahtarı kullanılarak bir şifreleme işlemi yapılacak olursa;
Şifreleme işlemi (*x* şifrelenecek materyal olmak üzere ve $x \in E,\ 0 \le k \le 12$)

$$Q = m \cdot P = 7 \cdot (2,\ 7) = (2,\ 7)$$

$k = 3$ şeklinde rasgele bir tamsayı seçilerek

$$e_K(x,\ k) = (k \cdot P,\ x + k \cdot Q)$$

$$\begin{aligned} y_1 &= k \cdot P = 3 \cdot (2,\ 7) \\ &= (8,\ 3) \end{aligned}$$

$$\begin{aligned} y_2 &= x + k \cdot Q = x + k \cdot m \cdot P \\ &= (10,\ 9) + 3 \cdot (7,\ 2) \\ &= (10,\ 2) \end{aligned}$$

$$y = (y_1,\ y_2) = ((8,\ 3),\ (10,\ 2))$$

şeklinde yapılabilir. Deşifreleme işlemi ise

$$\begin{aligned} x &= d_K(y_1,\ y_2) = y_2 - m \cdot y_1 \\ &= (10,\ 2) - 7 \cdot (8,\ 3) \\ &= (10,\ 2) - (3,\ 5) \\ &= (10,\ 2) + (3,\ 6) \\ &= (10,\ 9) \end{aligned}$$

$$x = (10,\ 9)$$

şeklinde yapılabilir.

7. SAYISAL İMZALAR

Bu bölümde, sayısal imzalar ve özellikle de eliptik eğriler kullanılarak gerçeklenen sayısal imzalar üzerinde durulacaktır.

7.1. Geleneksel İmzalar ve Sayısal İmzalar Arasındaki Farklar

Geleneksel imzalama metodu olan elyazısı ile atılan imzalar, bir belgeden sorumlu kişiyi belirtmek için kullanılır. Sayısal imzalar ise elektronik ortamda saklanan bir mesajın imzalanması işlemidir. Elektronik belge sayısal imza ile imzalandıktan sonra bilgisayar ağı üzerinde güvenli bir şekilde iletilebilecek duruma gelir.

Sayısal imzadaki ilk konu imzanın nasıl atıldığıdır. Geleneksel imzada fiziksel olarak var olan bir belgeye elyazısı ile imza atılarak işlem gerçekleştirilir. Sayısal imzada ise ilgili imza algoritmalar yardımı ile elektronik belgeye eklenir.

İmzanın doğrulanması sayısal imzalardaki bir başka önemli konudur. Geleneksel imza metodunda bu işlem, imzaların birbiri ile karşılaştırılması yoluyla yapılır. Elyazısı ile atılan imzaların kopyalanması ya da sahtesinin oluşturulması ihtimal dâhilinde olduğundan tam bir güvenlik söz konusu olamaz. Sayısal imzalarda ise imzanın doğrulanması geleneksel yöntemde yer alan karşılaştırma metodu yerine imza doğrulama algoritması kullanılarak yapılır. Herhangi bir kimse bu algoritmayı kullanarak imzanın doğruluğunu sorgulayabilir. Güvenli bir sayısal imza kullanımı bu imzanın sahtelerinin oluşturulmasının önüne geçilmesini sağlar.

Geleneksel imza ile sayısal imza arasındaki bir başka önemli fark da imzaların kopyaları konusudur. Sayısal imzalarda imzalanmış belgede bulunan imza orijinal imzanın aynısıdır. Ancak geleneksel imza kullanılarak imzalanmış bir belge ile orijinal imzalı belgedeki fark gözle görülebilir durumdadır. Bunun amacı, imzalanmış elektronik belgenin tekrar kullanımını önlemektir. Örneğin; *A* kişisi sayısal imza kullanarak *B* kişisine bir defalık kullanımı için bir hak veriyorsa, bu hakkın ikinci bir defa kullanılmasının engellenmesi gerekmektedir. Bu da sayısal imzaya bu tarz özellikler eklenmesi gereksinimini ortaya çıkarır (örneğin tarih, vs.).

Sayısal imza, sayısal imza oluşturma algoritması ve sayısal imza doğrulama algoritması olmak üzere iki temel bileşenden meydana gelir. *A* kişisi *x* mesajını gizli imza oluşturma algoritması $\mathcal{SIGN}$ kullanarak imzalar. Oluşturulan $\mathcal{SIGN}(x)$ imzası açık imza doğrulama algoritması $\mathcal{VER}$ kullanılarak herhangi biri tarafından doğrulanabilir. Bu doğrulama sonucunda imzanın doğruluğu kontrol edilmiş olur.

Daha önce de belirtildiği üzere sayısal imza birkaç bileşenden meydana gelir. Bu bileşenler ($\mathcal{P}$, $\mathcal{A}$, $\mathcal{K}$, $\mathcal{S}$, $\mathcal{V}$) ve şu şartları sağlarlar;

i) $\mathcal{P}$, sonlu sayıda olası mesajların kümesi

ii) $\mathcal{A}$, sonlu sayıda olası imzaların kümesi

iii) $\mathcal{K}$, anahtar uzayı, sonlu sayıda olası anahtarların kümesi

iv) Tüm $K \in \mathcal{K}$ olmak üzere, $\mathcal{SIGN}_K \in \mathcal{S}$ sayısal imza oluşturma algoritması ve buna karşılık gelen $\mathcal{VER}_K \in \mathcal{V}$ sayısal imza doğrulama algoritması mevcuttur. Tüm $\mathcal{SIGN}_K : \mathcal{P} \to \mathcal{A}$ ve $\mathcal{VER}_K : \mathcal{P} \times \mathcal{A} \to \{doğru, yanlış\}$ şeklinde mevcut olan fonksiyonlar tüm $x \in \mathcal{P}$ mesajları ve tüm $y \in \mathcal{A}$ imzaları için şu şartı sağlarlar;

$$\mathcal{VER}(x, y) = \begin{cases} doğru, & \text{eğer } y = \mathcal{SIGN}(x) \\ yanlış, & \text{eğer } y \neq \mathcal{SIGN}(x). \end{cases}$$

Tüm $K \in \mathcal{K}$ için $\mathcal{SIGN}_K$ ve $\mathcal{VER}_K$ fonksiyonları *polinom – zamanlı fonksiyonlardır.* (polynomial – time functions) $\mathcal{VER}_K$ fonksiyonu açık fonksiyon, $\mathcal{SIGN}_K$ gizli fonksiyondur. Verilen bir *x* mesajı için bu mesajın sahibi *A* kişisi dışındaki bir kişi tarafından sayısal imza atılması pratik olarak zor olmalıdır. *B* kişisi sayısal imza içeren bu mesajın doğruluğunu kontrol etmek istediği takdirde $\mathcal{VER}_K$ açık fonksiyonunu kullanır. Şayet bu fonksiyonun çıktısı *yanlış* ise, mesajın *A* kişisi tarafından imzalanmamış olduğu sonucuna ulaşılır [2].

7.2. Eliptik Eğri Sayısal İmza Algoritması

Eliptik Eğri Sayısal İmza Algoritması (Elliptic Curve Digital Signature Algorithm – ECDSA), 2000 yılında Amerikan Federal Bilgi İşleme Standartları tarafından onaylanmıştır (Federal Information Processing Standards – FIPS 186 – 2).

İmza oluşturma algoritmasına geçmeden önce imza oluşturma algoritmasında kullanılan parametrelerin açıklanması gerekmektedir. p bir asal sayıdır ve E eliptik eğrisi F_p cisminde tanımlıdır. A noktası ise, E üzerinde bulunan, q düzenine sahip ve çözümü zor ayrık logaritma problemi olan bir noktadır [2].

$\mathcal{P} = \{0,\ 1\}^*,\ \mathcal{A} = \mathbb{Z}_q^* \times \mathbb{Z}_q^*$ ise;

$\mathcal{K} = \{(p,\ q,\ E,\ A,\ m,\ B) : B = m \cdot A\},\ (0 \le m \le q-1)$

$p,\ q,\ E,\ A,\ B$ değerleri açık anahtarlar, m ise gizli anahtardır.

$K = (p,\ q,\ E,\ A,\ m,\ B)$ ve gizli rasgele k sayısı $(1 \le k \le q-1)$ için

$$\mathcal{SIGN}_K(x,\ k) = (r,\ s),$$

olur. Burada bulunan değerlerin hesaplanması ise şu şekildedir;

$k \cdot A = (u,\ v)$,
$r = u \pmod q$ ve
$s = k^{-1} \cdot (\mathcal{HASH}(x) + m \cdot r) \pmod q$.

Burada yapılan hesaplamalar sonucunda, $r = 0$ veya $s = 0$ değerleri bulunduğu takdirde yeni bir k değeri belirlenerek işlemler tekrarlanmalıdır.

Yukarıda açıklanan imza oluşturma algoritması ile oluşturulan imzanın doğruluğunun kontrol edilmesi için kullanılacak olan imza doğrulama algoritması ise

şu şekildedir; $x \in \{0, 1\}^*$ ve $r, s \in \mathbb{Z}_q^*$ için aşağıdaki işlemler yapılarak imzanın doğruluğu kontrol edilir.

$$w = s^{-1} \pmod{q}$$
$$i = w \cdot \mathcal{HASH}(x) \pmod{q}$$
$$j = w \cdot r \pmod{q}$$
$$(u, v) = i \cdot A + j \cdot B$$
$$\mathcal{VER}_K(x, (r, s)) = doğru \Leftrightarrow u \pmod{q} = r$$

Bahsedilen imza oluşturma ve imza doğrulama algoritmaları ile ilgili bir örnek vermemiz yerinde olacaktır.

7.1 Örnek

Z_{11}'de tanımlı E: $y^2 = x^3 + x + 6$ eğrisi üzerinde ve $p = 11$, $q = 13$, $A = (2, 7)$, $m = 7$ ve $B = (7, 2)$ parametreleri kullanılarak x düzmetni imzalanacaktır.

Bu imza oluşturma işlemi sırasında x düzmetnine $\mathcal{HASH}(x)$ şeklinde bir Hash Fonksiyonu uygulanması gerekmektedir. Bu örnekte $\mathcal{HASH}(x) = 4$ seçilerek imza oluşturma işlemine devam edilecektir.

$$(u, v) = k \cdot A = 3 \cdot (2, 7) = (8, 3),$$
$$r = u \pmod{q} = 8 \text{ ve}$$
$$s = k^{-1} \cdot (\mathcal{HASH}(x) + m \cdot r) \pmod{q} = 3^{-1} \cdot (4 + 7 \cdot 8) \pmod{13} = 7.$$

Yapılan işlemler sonucunda elde edilen imza (8, 7) şeklinde olacaktır. Yine biraz önce değinilmiş olan imza doğrulama algoritması kullanılarak doğrulama yapılacak olursa;

$w = s^{-1} \pmod{q} = 7^{-1} \pmod{13} = 2,$
$i = w \cdot HASH(x) \pmod{q} = 2 \cdot 4 \pmod{13} = 8,$
$j = w \cdot r \pmod{q} = 2 \cdot 8 \pmod{13} = 3,$
$(u, v) = i \cdot A + j \cdot B = 8 \cdot (2, 7) + 3 \cdot (7, 2) = (8, 3)$ ve
$u \pmod{q} = 8 = r.$
elde edilir.

7.2.1. Eliptik eğri sayısal imza algoritmasının büyük sayılara uygulanması

Şifreleme işlemlerinde, şifrelenmiş olan bilginin güvenliğinin sağlanması amacı ile büyük sayılar kullanılmaktadır. Gündelik hayatta kullanılan sayılar ile yapılan bir şifreleme işlemi, çeşitli hesaplama yöntemleri ve bilgisayarlar kullanılarak kolayca çözülebileceğinden dolayı büyük sayıların kullanımı büyük önem arz etmektedir.

Şifreleme işlemlerinde kullanılan sayılar genellikle 200 haneden büyük sayılardan seçilmektedir. Bu sayılar üzerinde işlem yapmak için ise çeşitli aritmetik algoritmalar kullanılmaktadır. Bahsedilmiş olan imza işlemlerinin büyük sayılar kullanılarak yapılması ile ilgili örnek vermemiz yerinde olacaktır.

Örnek. $p = 87435834987342897676478732747289063927131$, $a = 65327729$ ve $b = 121245$ olacak şekilde seçilerek Z_p'de tanımlı E: $y^2 = x^3 + 65327729x + 121245$ eğrisi kullanılacaktır. Bu eğri üzerinde bulunan noktalardan A = (6, 87404953778781179854559365621443212097732) noktası, sayısal imza işleminde kullanılacaktır. İmza oluşturulurken gerekli olan parametrelerden q = 5 ve m = 2 olarak seçildiğinde elde edilen B noktası ise şu şekilde olacaktır:

B = (51343219556084449348123322646289252392143,
27989419619474568880159628036791956070397)

Bu imza oluşturma işlemi sırasında x düzmetnine uygulanan Hash Fonksiyonu $HASH(x) = 4$ ve $k = 4$ olarak seçilmiştir.

$$(u,\ v) = k \cdot A = 4 \cdot (6,\ 874049537787811798545593656214432120977 32) = (2,\ 3),$$
$$r = u \pmod{q} = 2 \text{ ve}$$
$$s = k^{-1} \cdot (\mathcal{HASH}(x) + m \cdot r) \pmod{q} = 4^{-1} \cdot (4 + 2 \cdot 2) \pmod{5} = 2.$$

Yapılan işlemler sonucunda elde edilen imza (2, 2) şeklinde olacaktır. Yine biraz önce değinilmiş olan imza doğrulama algoritması kullanılarak doğrulama yapılacak olursa;

$$w = s^{-1} \pmod{q} = 2^{-1} \pmod{5} = 3,$$
$$i = w \cdot \mathcal{HASH}(x) \pmod{q} = 3 \cdot 4 \pmod{5} = 2,$$
$$j = w \cdot r \pmod{q} = 3 \cdot 2 \pmod{5} = 1,$$
$$(u,\ v) = i \cdot A + j \cdot B = 1 \cdot (6,\ 874049537787811798545593656214432120977 32) + 1 \cdot (51343219$$
$$5560844493481233226462892523921 43,\ 279894196194745688801596280367919560703 97)$$
$$= (2,\ 3) \text{ ve}$$
$$u \pmod{q} = 2 = r.$$

elde edilir.

8. ELİPTİK EĞRİ SAYISAL İMZA UYGULAMASI

Bu bölümde, tez kapsamında geliştirilmiş olan yazılım ile ilgili bilgiler verilecektir.

8.1. Yazılım Özellikleri

Günümüzde yazılım geliştirme işlerinde Nesne Yönelimli Programlama Teknikleri sıkça kullanılmaktadır. Tez kapsamında geliştirilmiş olan yazılım için de C++ Programlama Dili ve Nesne Yönelimli Programlama Teknikleri'nden olabildiğince fazla verim alınabilecek şekilde yararlanılmaya çalışılmıştır.

Eliptik eğrilerin oluşturulmasında ve eliptik eğriler üzerinde çok büyük sayılarla işlemler yapılabilmesi için iki adet farklı sınıf tanımlanmıştır. Programın çalışması esnasında bu sınıflar kullanılarak gerekli sayıda nokta ve büyük sayı nesneleri yaratılmaktadır. Bu sınıfların özelliklerine kısaca göz atacak olursak:

Class Verylong (sayı sınıfı): Açıklandığı üzere, kriptografik uygulamalarda yapılan işlemler sonucunda herhangi bir yuvarlama ya da sapma olmaması gerekmektedir. Ancak bilgisayarlar kullanılarak yapılan gündelik işlemlerde ve geliştirilen yazılımlarda yapılan işlemlerde, sonuçlar belli bir sınırdan sonra yuvarlanmaktadır. Bunun önüne geçmek amacıyla *verylong sınıfı* tanımlanmıştır. Bu sınıf tanımlanırken, sayının basamak sayısı ve oluştuğu rakamlar temel özellikler olarak alınmış ve normal sayılarla yapılabilmekte olan dört işlem ve mantıksal karşılaştırmalar gibi işlemler bu sınıf içerisinde de tekrar tanımlanmıştır. Ayrıca, yapılmakta olan işlemlerden daha çabuk sonuç elde edilmesi amacı ile birçok farklı algoritma da bu sınıfa dahil edilmiştir.

Class Point (nokta sınıfı): Eliptik eğriler kullanılarak işlem yapılırken karşılaşılabilecek olan problemleri en aza indirmek amacı ile yeni bir *nokta sınıfı* tanımlanmıştır. Bu sınıf tanımlanırken daha çok eliptik eğrilere yönelik özellikler gözönünde bulundurulmuş ve buna yönelik fonksiyonlar geliştirilmeye çalışılmıştır. Bu sınıf içerisinde, noktalar arası toplama işlemleri, nokta katı alma işlemleri ve bu

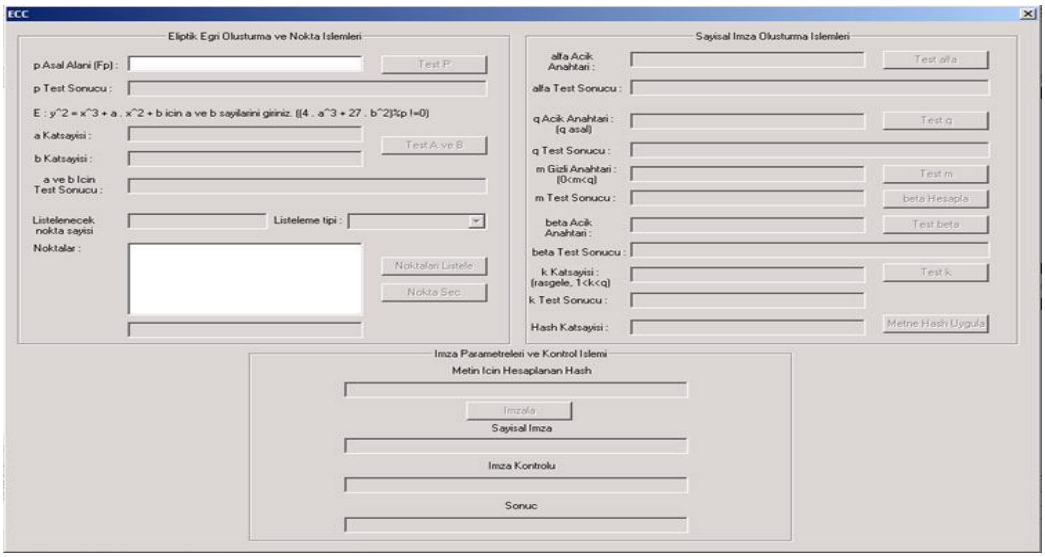
gibi birçok fonksiyon tanımlanmıştır. Oluşturulmuş olan büyük sayı sınıfı ile nokta sınıfı birbirleri ile entegre olabilecek şekilde tasarlanmıştır.

Oluşturulmuş olan bu sınıfların yanı sıra sınıflardan bağımsız olan ancak sınıflar tarafından da direk olarak kullanılabilen birçok fonksiyon tanımlanmıştır. Euclid Algoritması, Legendre Sembol Hesaplaması ve bu tarz birçok matematiksel algoritma program içerisinde yazılmıştır. Sınıf tanımları Ek – 1'de görülebilir.

Metinlerin eliptik eğri sayısal imza sistemi kullanılarak işleme sokulabilmesi için, imzalanacak olan metinlere hash algoritması uygulanması gerekmektedir. Bu amaçla serbest bir yazılım, programla entegre çalışabilecek şekilde değiştirilmiş ve bazı temel özellikleri aynı kalmak üzere, yeniden tasarlanmıştır. Programın kullanıcı ile etkileşimde bulunulması amacı ile oluşturulmuş arayüzü tasarlanırken Microsoft Temel Sınıfları (Microsoft Foundation Classes-MFC) kullanılmış ve program Microsoft Visual C++ 6.0 geliştirme ortamında ve Microsoft Windows XP Professional işletim sistemi üzerinde geliştirilmiştir.

8.2. Yazılım Arayüzü ve Kullanımı

Program ilk çalıştırıldığında Şekil 8.1'deki gibi bir arayüzle karşılaşılacaktır.



Şekil 8.1 Eliptik eğri sayısal imza uygulaması ekran görüntüsü

Program çalıştırıldığında kullanıcıdan eliptik eğrinin tanımlı olacağı ve işlemlerin yapılacağı cismin belirlenmesi için asal olan *p* değerini girmesi gerekmektedir. Bu değerin girilmesinin ardından eliptik eğri oluşturulmasında kullanılacak olan *a* ve *b* değerleri girilmelidir. Şayet belirlenen parametreler ile ilgili yapılan testler neticesinde herhangi bir problem ile karşılaşılmazsa program tarafından eliptik eğri oluşturulmuş durumda olacaktır. Bu aşamadan sonra kullanıcıya, ilgili cisim ve eliptik eğri üzerinde tanımlı noktalar listelenecektir. Kaç adet nokta listeleneceği ve noktaların aralığı gibi özellikler de program üzerinden ayarlanabilmektedir. İlgili seçimler yapıldıktan sonra kullanılacak olan *alfa noktası* seçilerek programın ilk adımı tamamlanmış olur.

Eliptik eğri oluşturma ve nokta seçme işlemleri tamamlandıktan sonra, imza işleminde kullanılacak olan parametrelerin sırasıyla ve belirtilen şartlara göre girilmesi gerekmektedir. Kullanıcı tarafından girilen *q* ve *m* katsayılarına göre *E* eliptik eğrisi üzerinde *beta noktası* hesaplanacak ve bu nokta sayısal imza işleminde kullanılacaktır. Kullanıcının son olarak rasgele bir *k* katsayısı girmesi gerekmektedir. Bu aşamadan sonra konumu belirtilen imzalanacak olan metne *hash algoritması* uygulanacaktır. Şayet metin imzalanmak istenmiyorsa burada herhangi bir değer de kullanılabilir.

Tüm değerler uygun olarak girildikten ve metne hash algoritması uygulandıktan sonra “İmzala” tuşu kullanılarak metin imzalanabilir ve imza doğrulama işlemleri ilgili alanlardan izlenebilir.

9. SONUÇ

Günlük yaşam içerisinde elektronik ortamda yapılan birçok işlemde bilgi güvenliği büyük önem arz etmektedir. Kullanılan sistemlerin ve güvenlik önlemlerinin karmaşık yapısına rağmen bu öğeler üzerinde her zaman çeşitli açıklar ve zayıf noktalar bulunmaktadır. Karşılaşılan bu problemlerden dolayı ise yapılmış olan işlemlerin güvenliği ve neticesinde de kişisel bilgilerin gizliliği konusu her zaman varlığını korumaktadır.

Güvenlik problemleri neticesinde sistemlerin özellikleri geliştirilirken çeşitli şifreleme sistemleri de sıkça kullanılmakta ve kullanılan bu sistemler devamlı bir yenilenme sürecinde olmaktadır. Bu yenilenme sürecinin ortaya çıkardığı en önemli sistemlerden biri de hiç şüphesiz ki Açık Anahtarlı Şifreleme Sistemleri'dir. Bu sistemlerden başlıcası olan ve günümüzde de popüler olarak kullanılmakta olan RSA Şifreleme Sistemi ile ilgili kimi uygulama zorlukları ile yaşanmasından dolayı yeni sistemlerin tasarlanması ve uygulamaya sokulması zorunluluğu ortaya çıkmıştır.

Bir diğer Açık Anahtarlı Şifreleme Sistemi olan Eliptik Eğri Şifreleme Sistemi de diğer sistemlerde yaşanan çeşitli problemler neticesinde geliştirilmiştir. Bu sistem ile ilgili çeşitli konular halen tartışılmakta ise de yapılan birçok çalışmanın ardından sistem kullanılmaya başlanmış ve diğer sistemlere nazaran sağladığı uygulama kolaylığı ve yüksek güvenlik sayesinde de kendisine birçok kullanım alanı bulmuştur.

Eliptik Eğri Sayısal İmza Sistemleri de temelinde yatan eliptik eğri şifreleme algoritmaları sayesinde daha büyük önem kazanmaya ve kullanılırlığını artırmaya devam edecektir.

KAYNAKLAR

1. Hankerson, D., Mezenes, A. J., Vanstone, S. A., "Guide to Elliptic Curve Cryptography", 1st ed., ***Springer – Verlag***, New York, 25 – 152 (2004).

2. Stinson, D. R., "Cryptography Theory and Practice", 2nd ed., ***Chapman & Hall/CRC***, Florida, 173 – 180, 274 – 292 (2002).

3. Washington, L. C., "Elliptic Curves Number Theory and Cryptography", 1st ed., ***Chapman & Hall/CRC***, Florida, 89 – 175 (2003).

4. Knuth, D. E., "The Art of Computer Programming, Vol. 2", 2nd ed., ***Addison – Wesley Publishing Company***, Amerika, 250 – 278 (1981).

5. Koblitz, N., "A Course in Number Theory and Cryptography", 2nd ed., ***Springer-Verlag***, New York, 1527-196 (1987).

6. Johnson, D., Mezenes, A. J., "The Elliptic Curve Digital Signature Algorithm (ECDSA)", ***University of Waterloo***, ***Technical Report CORR 99-34***, Waterloo, 11 – 26 (2000).

7. Şenay, H., "Sayılar Teorisine Giriş", 1ci basım, ***Selçuk Üniversitesi Yayınları***, Konya, 1 – 49 (1989).

8. Mezenes, A. J., van Oorschot, P. C., Vanstone, S. A., "Handbook of Applied Cryptography", 5th ed., ***CRC Press***, Florida, 49 – 132 (2001).

9. Schneier, B., "Applied Cryptography", 2nd ed., ***John Wiley & Sons***, Amerika, 381 – 419 (1996).

10. Afacan, E., Değirmenci F., "Çok Büyük Sayılar, Şifreleme Teknikleri ve RSA", Kara Harp Okulu, ***13 Mart Bilim, Kültür ve Spor Etkinlikleri***, Ankara, 13 – 65 (2006).

11. Şenkon, H., "Soyut Cebir Dersleri", 1ci basım, ***İstanbul Üniversitesi Fen Fakültesi Basımevi***, İstanbul, 72 – 80 (1990).

12. Giblin, P., "Primes and Programming", 1st ed., ***Cambridge University Press***, Cambridge, 85 – 112 (1993).

13. Apostol, T. M., "Introduction to Analytic Number Theory", 1st ed., ***Springer – Verlag***, New York, 257 – 302 (1976).

14. Lafore, R., "Object Oriented Programming with C++", 2nd ed., ***Sams Publishing***, New York, 228 – 256 (2002).

15. Aslan, K., "A'dan Z'ye C Kılavuzu", 2ci basım, ***Pusula Yayıncılık ve İletişim Ltd. Şti.***, İstanbul, 310 – 355 (1998).

16. Çallıalp, F., "Soyut Matematik", ***Beta Basım Yayın***, İstanbul (1995).

EKLER

EK-1. Eliptik eğri sayısal imza uygulaması sınıf tanımları

Eliptik Eğri Sayısal İmza Uygulaması için geliştirilmiş olan sınıflar aşağıda görülmektedir.

```
class verylong
{
private :
char vlstr[SZ];
int vlen;
public :
verylong( ) : vlen( 0 )
{
vlstr[0] = '\0';
}
verylong( const char s[SZ] )
{
strcpy( vlstr, s );
vlen = strlen( s );
}
verylong( const unsigned long n )
{
ltoa( n, vlstr, 10 );
vlen = strlen( vlstr );
}
~verylong( )
{
}
void putvl( ) const;
void getvl( );
verylong operator + ( const verylong );
verylong operator - ( const verylong );
```

EK-1. (Devam) Eliptik eğri sayısal imza uygulaması sınıf tanımları

```
verylong operator * ( const verylong );
verylong operator / ( const verylong );
verylong operator % ( const verylong );
int operator > ( const verylong );
int operator >= ( const verylong );
int operator < ( const verylong );
int operator <= ( const verylong );
int operator == ( const verylong );
int operator != ( const verylong );
int normalize( int[] , int );
int  unnormalize( verylong, int[], int );
verylong powerc( verylong , verylong );
void savefile( int );
verylong sqrmltply( verylong, verylong, verylong );
CString vlToString( verylong );
void setContent(CString);
};

class point
{
private:
verylong mcoordinateX;
verylong mcoordinateY;
public:
point()
{
mcoordinateX = "0";
mcoordinateY = "0";
}
point(verylong X, verylong Y)
```

EK-1. (Devam) Eliptik eğri sayısal imza uygulaması sınıf tanımları

```
{
mcoordinateX = X;
mcoordinateY = Y;
}
~point()
{
}
void showPoint();
int isExist();
verylong calcSlope(point, point);
verylong calcSlope(point);
verylong inverse(verylong, verylong);
point operator + (point);
point exponent(point, verylong);
verylong adding(point, point, verylong);
verylong adding(point, point, verylong, verylong);
verylong doublingXp(point, verylong);
verylong doublingYp(point, verylong, verylong);
verylong getCoordinateX(point);
verylong getCoordinateY(point);
};
```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	: DEĞİRMENCİ, Fatih
Uyruğu	: T.C.
Doğum tarihi ve yeri	: 20.09.1978, Bursa
Medeni hali	: Bekar
Telefon	: 00353-86-245 21 39
Faks	: –
e-mail	: fdegir@gmail.com.

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Lisans	Gazi Üniversitesi / Elektrik ve Elektronik Mühendisliği Bölümü	2004
Lise	Eskişehir Fatih Fen Lisesi	1995

İş Deneyimi

Yıl	Yer	Görev
2006 – DEVAM	LM Ericsson Limited, İrlanda	Yazılım Mühendisi
2004 – 2006	Havelsan A.Ş.	Yazılım Mühendisi

Yabancı Dil

İngilizce, Almanca.

Hobiler

Bilgisayar teknolojileri, matematik, kitap okumak.