AYRIK LOGARİTMA PROBLEMİ

Semiha TURP

T.C.
BURSA ULUDAĞ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AYRIK LOGARİTMA PROBLEMİ

Semiha TURP
https://orcid.org/0000-0002-9851-2009

Doç. Dr. Betül GEZER

YÜKSEK LİSANS TEZİ
MATEMATİK ANABİLİM DALI

BURSA – 2019

TEZ ONAYI

Semiha TURP tarafından hazırlanan "Ayrık Logaritma Problemi" adlı tez çalışması aşağıdaki jüri tarafından oy birliği ile Bursa Uludağ Üniversitesi Fen Bilimleri Enstitüsü Matematik Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. Betül GEZER
https://orcid.org/0000-0001-9133-1734

Başkan : Prof. Dr. Osman BİZİM
https://orcid.org/0000-0001-5236-4023
Bursa Uludağ Üniversitesi, Fen Edebiyat Fakültesi, Matematik Anabilim Dalı

Üye : Doç. Dr. Betül GEZER
https://orcid.org/0000-0001-9133-1734
Bursa Uludağ Üniversitesi, Fen Edebiyat Fakültesi, Matematik Anabilim Dalı

Üye : Prof. Dr. Ahmet YILDIZ
https://orcid.org/0000-0002-9799-1781
İnönü Üniversitesi, Eğitim Fakültesi, Matematik ve Fen Bilimleri Eğitimi Anabilim Dalı

Yukarıdaki sonucu onaylarım

Prof. Dr. Hüseyin Aksel EREN
Enstitü Müdürü
.../.../....

B.U.Ü. Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

27/09/2019

Semiha TURP

ÖZET

Yüksek Lisans Tezi

AYRIK LOGARİTMA PROBLEMİ

Semiha TURP

Bursa Uludağ Üniversitesi
Fen Bilimleri Enstitüsü
Matematik Anabilim Dalı

Danışman: Doç. Dr. Betül GEZER

Bu çalışmada ayrık logaritma problemi ve bu problemin çözümleri ele alınmış ve eliptik eğri ayrık logaritma problemini daha kolay bir ayrık logaritma problemine dönüştüren algoritmalar verilmiştir.

Birinci bölümünde cebir ve sayılar teorisi ile ilgili temel kavramlar verildikten sonra kriptoloji ile ilgili temel kavramlar üzerinde durulmuştur.

İkinci bölümde ayrık logaritma problemi ve bu problemin çözümünde kullanılan çeşitli algoritmalar ele alınmıştır. İlk olarak Diffie ve Hellman anahtar değişimi algoritması ele alınmış ve El-Gamal açık anahtar kriptosistemleri üzerinde durulmuştur. Daha sonra problemin çözümü için çeşitli algoritmalar verilmiştir.

Üçüncü bölümde eliptik eğriler ve eliptik eğri ayrık logaritma problemi ele alınmıştır. Bu bölümde ise eliptik eğri ayrık logaritma problemini bir ayrık logaritma problemine dönüştüren algoritmalar verilmiştir. Dördüncü bölümde ise bir eliptik eğrinin bölüm polinomları kavramı kullanılarak benzer algoritmalar verilmiştir.

Anahtar Kelimeler: Açık anahtar kriptolojisi, ayrık logaritma problemi, eliptik eğriler.
2019, vii + 85 sayfa.

ABSTRACT

MSc Thesis

THE DISCRETE LOGARITHM PROBLEM

Semiha TURP

Bursa Uludağ University
Graduate School of Natural and Applied Sciences
Department of Mathematics

Supervisor: Doç. Dr. Betül GEZER

In this work, the discrete logarithm problem and solutions of this problem are disscused and the algorithms are given to reduce the elliptic curve discrete algorithm problem to an easier discrete logartihm problem.

In the first chapter, some fundamental concepts on the theory of algebra and number theory and cryptography are given.

In the second chapter, the discrete logarithm problem and the algorithms that used for the solutions of this problem are disscused. Firstly, Diffie and Hellman key exchange algorithm is considered and the El-Gamal public key cryptosystem is discussed. Then some algorithms are given for solving the discrete logarithm problem.

In the third chapter, elliptic curves and elliptic curve discrte logarithm problem are considered. In this chapter, some algorithms are given to reduce the elliptic curve discrete algorithm problem to a discrete logartihm problem. In the fourth chapter, similar algorithms are given by using the division polynomials of an elliptic curve.

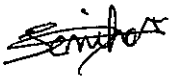
Key words: The public key cryptology, the discrete logarithm problem, elliptic curves.
2019, vii + 85 pages.

TEŞEKKÜR

Yüksek Lisans çalışmamın her aşamasında bilgi ve tecrübelerinden yararlandığım, hoşgörüsüyle her zaman yanımda olan saygıdeğer hocam Doç. Dr. Betül GEZER'e teşekkür ederim.

Ayrıca bu tez çalışması boyunca bana her türlü manevi desteği veren aileme de teşekkürü bir borç bilirim.

Semiha TURP
27/09/2019



İÇİNDEKİLER

Sayfa

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler	**Açıklama**
K	anahtarların uzayı
der(D)	bir D böleninin derecesi
sum(D)	bir D böleninin toplamı
$\boldsymbol{O}$	bir eliptik eğri üzerindeki sonsuzdaki nokta
$E[m]$	bir E eliptik eğrisinin m-büküm alt grubu
e_m	bir eliptik eğri üzerindeki Weil eşleştirmesi
$E(\mathbb{F})$	bir $\mathbb{F}$ cismi üzerinde tanımlı E eliptik eğrisi üzerindeki noktalar grubu
$\boldsymbol{\mu}_m$	birimin m. ilkel köklerinin grubu
$O(g(x))$	büyük-O gösterimi
M	düz metin mesajlarının uzayı
div(f)	f fonksiyonunun böleni
t_p	Frobenius endomorfizminin izi
$\log_g(h)$	g tabanına göre h elemanının ayrık logaritması
$\mathbb{Z}_m$	m modülüne göre kalan sınıflarının kümesi
$\mathbb{F}_p$	p bir asal sayı olmak üzere p elemanlı sonlu cisim
$\mathbb{F}_p^k$	p bir asal sayı olmak üzere p^k elemanlı sonlu cisim
$\log_P(Q)$	Q noktasının P noktasına göre eliptik ayrık logaritması
C	şifreli metinlerin uzayı
e veya e_k	şifreleme fonksiyonu
d veya d_k	şifre çözme fonksiyonu
$\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$	tamsayılar, rasyonel sayılar, gerçel ve karmaşık sayılar kümeleri
$\pi(X)$	2 ve X arasındaki asal sayıların sayısı
$\psi(X, B)$	2 ve X arasındaki B-düzgün tamsayıların sayısı

Kısaltmalar	**Açıklama**
ALP	Ayrık logaritma problemi
DHP	Diffie-Hellman problemi
EEALP	Eliptik eğri ayrık logaritma problemi
MOV	Menezes, Okamoto, Vanstone
RSA	Rivest, Shamir ve Adleman
SEA	Schoof, Elkies, Atkin

ŞEKİLLER DİZİNİ

Sayfa

ÇİZELGELER DİZİNİ

Sayfa

1. GİRİŞ

Bu bölümde ayrık logaritma problemi teorisinde kullanılacak temel kavramlar ve teoremler ele alınacaktır.

1.1. Cebir ve Sayılar Teorisi ile İlgili Temel Kavramlar

Bu kısımda kriptolojide kullanılan cebir ve sayılar teorisi ile ilgili bazı temel kavram ve teoremler ele alınacaktır. İlk olarak kriptolojide sonlu cisimlerin kullanılması durumunda karşılaşılan temel kavramlar ele alınacak, daha sonra modül kavramı üzerinde durulacaktır.

Aşağıda sayılar teorisinin en temel teoremlerinden biri verilmektedir.

1.1.1. Fermat'nın Küçük Teoremi. p bir asal sayı ve a herhangi bir tamsayı olmak üzere

$$a^{p-1} \equiv \begin{cases} 1 \pmod{p}, & p \nmid a \\ 0 \pmod{p}, & p \mid a \end{cases}$$

dir (Fraleigh 2003).

1.1.2. Uyarı. Kısım 1.2 ele alınacak olan hızlı kuvvet alma algoritması ve Teorem 1.1.1, herhangi bir $a \in \mathbb{F}_p$ sayısının p modülüne göre tersinin hesaplanmasında oldukça kolaylık sağlar. Gerçektende, Fermat'nın Küçük Teoremi gereği, a^{p-2} sayısı a sayısının çarpımı p modülüne göre 1'e denk olduğundan

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

dir.

Fermat'nın Küçük Teoremi'ne göre, $p \nmid a$ ise $a^{p-1} \equiv 1 \pmod{p}$ dir. Bununla birlikte, herhangi bir a tamsayısının p modülüne göre 1'e denk olan daha küçük kuvvetleri bulunabilir. $a^k \equiv 1 \pmod{p}$ olacak şekilde en küçük $k \geq 1$ tamsayısına a sayısının p modülüne göre *mertebesi* denir.

1.1.3. Önerme. p bir asal sayı, a bir tamsayı ve $p \nmid a$ olmak üzere $a^n \equiv 1 \pmod{p}$ olsun. Bu durumda a sayısının p modülüne göre mertebesi n sayısını böler. Özel olarak a sayısının mertebesi $p - 1$ sayısını böler (Fraleigh 2003).

Cebir ve sayılar teorisinden de hatırlanacağı gibi, $\mathbb{F}_p$ bir sonlu cisim ise $\mathbb{F}_p^* = \mathbb{F}_p \setminus \{0\}$ grubu bir devirli gruptur. Aşağıda bu devirli grubun üretecine özel bir isim verilecektir.

1.1.4. Tanım. $\mathbb{F}_p^*$ devirli grubunu üreten $g \in \mathbb{F}_p^*$ elemanlarına $\mathbb{F}_p$ sonlu cisminin *ilkel kökleri* denir.

Tanıma dikkat edilirse $g \in \mathbb{F}_p^*$ bir ilkel kök ise g elemanının mertebesi $p - 1$ dir ve

$$\mathbb{F}_p^* = \{1, g, g^2, g^3, \ldots, g^{p-2}\}$$

dir.

Bilindiği üzere her bir abelyen G grubuna $\mathbb{Z}$ üzerinde bir "vektör uzayı" olarak bakılabilir. Vektör uzayı tanımındaki $\mathbb{F}$ cisminin yerine bir R halkasının alınması halinde elde edilen cebirsel yapıya "*R halkası üzerinde bir vektör uzayı*" denir veya bu ifade yerine "*R halkası üzerinde bir modül*" ifadesi kullanılır. Buradan anlaşıldığı gibi modül kavramı vektör uzayı kavramının bir genellemesidir. Vektör uzayı ile ilgili kavramların birçoğu modüller için de geçerlidir. Bu nedenle modüller de cebir ve matematiğin birçok dalında kullanılan önemli cebirsel yapılarından birisidir.

1.1.5.Tanım. R bir halka, $(M, +)$ bir değişmeli grup ve

$$\cdot : R \times M \to M$$

olmak üzere, her $r, s \in R$ ve her $\alpha, \beta \in M$ için

M1. $r\alpha \in M$,

M2. $r(\alpha + \beta) = r\alpha + r\beta$,

M3. $(r + s)\alpha = r\alpha + s\alpha$,

M4. $(rs)\alpha = r(s\alpha)$

koşulları gerçekleniyorsa $({}_R M, +, \cdot)$ cebirsel yapısına bir *sol R- modül* denir ve kısaca ${}_R M$ ile gösterilir. Eğer R birimli halka ve her $\alpha \in M$ için $1\alpha = \alpha$ ise ${}_R M$ modülüne bir *birimli sol R-modül* denir.

1.1.6. Tanım. M ve N, R-modül olmak üzere $T: M \to N$ dönüşümü her $m, n \in M$ ve her $r \in R$ için

$$T(m + n) = T(m) + T(n), \qquad T(rm) = rT(m)$$

koşullarını gerçekliyorsa T dönüşümüne bir R-modül homomorfizmi veya kısaca R-homomorfizmi denir.

1.1.7. Uyarı 1. M ve N birer vektör uzayı, yani R bir cisim ise R-modül homomorfizmine *R-lineer dönüşüm* denir.

2. Bundan başka her $m, n \in M$ ve her $r, s \in R$ için,

$$T(rm + sn) = r\,T(m) + s\,T(n)$$

koşulu gerçekleniyorsa T dönüşümü bir R-modül homomorfizmidir.

1.1.8. Tanım. M bir R-modül ve $S \subset M$ olsun. $r_i \in R$ ve $m_i \in S$ olmak üzere

$$r_1m_1 + \ldots + r_km_k$$

biçimindeki tüm lineer birleşimlerin ailesi S kümesini bulunduran en küçük alt modüldür. Bu modüle $S \subset M$ *kümesi ile üretilen alt modül* denir ve bu küme $<S>$ ile gösterilir.

Eğer $<S> = M$ ise M R-modülüne S kümesi ile üretilmiş modül veya S kümesine M R-*modülünün üreteç kümesi* denir. Eğer M R-modülünün üreteç kümesi sonlu bir küme ise M R-modülüne *sonlu üreteçli modül* denir.

1.1.9. Tanım. S kümesi M R-modülünün bir üreteç kümesi ve üstelik S kümesi lineer bağımsız ise S kümesine M R-*modülünün bir bazı* ve bir bazı olan M R-modülüne de *serbest modül* denir.

1.2. Hızlı Kuvvet Algoritması

RSA ve Diffie-Hellman gibi bazı şifreleme sistemlerinde bir g sayısının modülo N ye göre çok büyük kuvvetlerinin hesaplanması gerekmektedir. Burada N, yüzlerce basamağı olan bir sayı olabilir. A herhangi bir sayı olmak üzere, g^A değerini hesaplamanın bir yolu bulunan her sayıyı g sayısı ile tekrar çarpmaktır. Böylece

$$g_1 \equiv g \pmod N, \quad g_2 \equiv g \cdot g_1 \pmod N, \quad g_3 \equiv g \cdot g_2 \pmod N, \quad g_4 \equiv g \cdot g_3 \pmod N, \quad \ldots$$

dir. Buna göre, $g_A \equiv g^A \pmod N$ dir, ancak A çok büyük bir sayı, örneğin, $A \approx 2^{1000}$ ise bu yöntemle hesapla yapmak dünyanın yaşından daha fazla zaman alır. Bu nedenle $g^A \pmod N$ değerini hesaplamak için kullanışlı ve daha hızlı bir yol kullanmak gerekir. Aşağıda ele alınacak hızlı kuvvet algoritmasına göre, N modülüne göre g^A sayısını, arka arkaya kare alma ve çarpma işlemleri kullanarak hesaplamak için A sayısının ikili açılımı kullanır. Bu algoritmayı vermeden önce aşağıdaki örneği inceleyelim.

1.2.1. Örnek. $3^{218} \pmod{1000}$ değerini hesaplamak için 218 sayısı 2 nin kuvvetlerinin bir toplamı olarak

$$218 = 2 + 2^3 + 2^4 + 2^6 + 2^7$$

biçiminde yazılabilir. Böylece 3^{218} sayısı

$$3^{218} = 3^{2+2^3+2^4+2^6+2^7} = 3^2 \cdot 3^{2^3} \cdot 3^{2^4} \cdot 3^{2^6} \cdot 3^{2^7}$$

olarak yazılabilir. Dikkat edilirse 3^2, 3^{2^2}, 3^{2^3}, 3^{2^4}, … dizisindeki değerlerin her biri bir önceki değerin karesi olduğundan bu dizideki her bir terimi hesaplamak oldukça kolaydır. Üstelik bu değerler modülo 1000'e göre hesaplanacağından üç basamaktan daha fazla depolama alanına ihtiyaç yoktur. Aşağıdaki tabloda modülo 1000 e göre 3 sayısının 3^{2^7} kadar olan kuvvetleri hesaplanmıştır. $3^{2^7} = 3^{218}$ sayısının büyük bir kuvveti olduğu halde her bir değer bir önceki değerin karesi olduğundan bu tabloyu oluşturmak için sadece yedi tane çarpma işlemi yapılmıştır.

Çizelge 1.1. 1000 modülüne göre 3 sayısının kuvvetleri (Hoffstein ve ark. 2008)

i	0	1	2	3	4	5	6	7
$3^{2^i} \pmod{1000}$	3	9	81	561	721	841	281	961

Yukarıdaki çizelgedeki değerlerden ihtiyaç duyulanlar kullanarak

$$\begin{aligned} 3^{218} &\equiv 3^2 \cdot 3^{2^3} \cdot 3^{2^4} \cdot 3^{2^6} \cdot 3^{2^7} \\ &\equiv 9 \cdot 561 \cdot 721 \cdot 281 \cdot 961 \pmod{1000} \\ &\equiv 489 \pmod{1000} \end{aligned}$$

olarak bulunur.

Dikkat edilirse $9 \cdot 561 \cdot 721 \cdot 281 \cdot 961$ çarpma işlemi yapılırken her çarpma işleminden sonra 1000 modülüne göre indirgeme yapılarak büyük sayıların oluşumu engellenmiş ve $3^{218} \pmod{1000}$ değeri hesaplanırken sadece 11 tane çarpma işlemi yapılmıştır.

Bu yöntem *Hızlı Kuvvet Algoritması* veya *Kare ve Çarpım Algoritması* olarak adlandırılır.

1.2.2. Algoritma.

1. $A_r = 1$ olduğunu varsayalım ve $A_0, A_1, A_2, \ldots, A_r \in \{0, 1\}$ olmak üzere A sayısının

$$A = A_0 + A_1 \cdot 2 + A_2 \cdot 2^2 + A_3 \cdot 2^3 + \ldots + A_r \cdot 2^r$$

ikili açılımını hesapla.

2. $0 \le i \le r$ için ardışık kare alma işlemi yaparak $g^{2^i} \pmod N$ değerlerini, yani

$$\begin{aligned} a_0 &\equiv g \pmod N, \\ a_1 &\equiv a_0^2 \equiv g^2 \pmod N, \\ a_2 &\equiv a_1^2 \equiv g^{2^2} \pmod N, \\ &\vdots \\ a_r &\equiv a_{r-1}^2 \equiv g^{2^r} \pmod N \end{aligned}$$

değerlerini hesapla.

3. $g^A = g^{A_0 + A_1 \cdot 2 + A_2 \cdot 2^2 + \cdots + A_r \cdot 2^r} = g^{A_0} \, (g^2)^{A_1} \cdot (g^{2^2})^{A_2} \cdot (g^{2^3})^{A_3} \cdots (g^{2^r})^{A_r}$

$\equiv a_0^{A_0} \cdot a_1^{A_1} \cdot a_2^{A_2} \cdot a_3^{A_3} \cdots a_r^{A_r} \pmod N$

eşitlik ve denkliği kullanarak $g^A \pmod N$ değerini hesapla.

Dikkat edilirse $a_0, a_1, \ldots, a_r$ değerleri 2. adımda hesaplanmışlardır, doğal olarak $A_0, \ldots, A_r \in \{0, 1\}$ olduğundan bu çarpımda sadece kuvvetleri 1 olan a_i değerleri dikkate alındığından fazladan en çok r tane daha çarpma işlemi söz konusudur. Dolayısıyla $g^A \pmod N$ değerini hesaplamak için en çok $2r$ tane çarpma işlemi gerekir. $A \ge 2^r$ olduğundan $g^A \pmod N$ değerini hesaplamak için en çok $2\log_2(A)$ çarpma işlemi yapıldığı görülür. Eğer A sayısı çok büyük bir sayı, örneğin, $A \approx 2^{1000}$ ise bir bilgisayarın yaklaşık 2000 çarpma işlemi yaparak $2^A \pmod N$ değerini hesaplaması oldukça kolaydır.

1.3. Simetrik ve Asimetrik Şifreler

Bir *A* kişisinin bir *B* kişisine gizli bir mesaj göndermek istediğini varsayalım. *A* kişisi *m* düz metin mesajını şifreleyerek bir *c* şifreli metnine çevirmek için gizli bir *k* anahtarı kullanır. *B* kişisi *c* şifreli metnini aldıktan sonra bu metni deşifre etmek ve *m* düz metin mesajını yeniden oluşturmak için *k* gizli anahtarını kullanır. Eğer bu yöntem düzgün çalışıyorsa *A* ve *B* kişileri *k* gizli anahtarına sahiptirler. Üstelik bu sistem güvenli ise *A* ve *B* kişileri dışında başka bir *C* kişisinin *k* gizli anahtarını bilmemesi, tahmin edememesi ve *k* gizli anahtarını bilmeden *c* şifreli metninden *m* düz metnini elde edememesi gerekir.

Bu kısımda "*kriptosistem*" kavramı soyut matematiksel terimlerle ifade edilecektir. Böylece değişik sistemler arasındaki benzerlikler ve farklılıklar vurgulanırken aynı zamanda bir şifreleme sisteminin çeşitli saldırılara karşı güvenliği incelenecektir.

Simetrik Şifreler

Simetrik şifreleme yönteminde *A* ve *B* kişilerinin gizli *k* anahtarını paylaşmaları gerekir. Çünkü bu gizli anahtarı kullanarak mesajları hem şifreleyip hem de şifrelerini çözebilirler. Böylece *A* ve *B* kişileri eşit (veya simetrik) bilgi ve yeteneklere sahip olurlar. Bu nedenle bu tür şifrelere *simetrik şifreler* adı verilir.

Matematiksel olarak, bir simetrik şifre olası anahtarların bulunduğu *K* uzayından bir *k* anahtarı seçerek olası mesajların bulunduğu *M* uzayından seçilen bir düz *m* mesaj metnini şifreler. Dolayısıyla düz *m* metninin *k* anahtarı ile olası şifrelenmiş mesajların bulunduğu *C* uzayından bir *c* şifreli mesajı elde edilmiş olur.

Böylece $K \times M$, *k* anahtarı ve *m* düz metninden oluşan (*k*, *m*) çiftlerinin ve *C*, şifreli metinlerin kümesi olmak üzere simetrik şifreleme işlemi

$$e : K \times M \rightarrow C$$

biçiminde bir fonksiyon olarak düşünülebilir. Benzer şekilde şifreli metni çözme fonksiyonu da

$$d : K \times C \rightarrow M$$

biçiminde tanımlanır. Dolayısıyla c şifreli metnini çözerek m düz metni elde edilmek istendiğinden bu durum matematiksel olarak her $k \in K$, her $m \in M$ için $d(k, e(k, m)) = m$ biçiminde ifade edilir.

Genellikle bu işlemler, k bir alt indis olarak alınarak her bir k anahtarı ve her $m \in M$ için $d_k(e_k(m)) = m$ olacak biçimde

$$e_k : M \rightarrow C \quad \text{ve} \quad d_k : C \rightarrow M$$

fonksiyonları ile ifade edilir. Dikkat edilirse her k anahtarı için d_k fonksiyonu e_k fonksiyonunun tersidir, $e_k(m) = e_k(m')$ ise

$$m = d_k(e_k(m)) = d_k(e_k(m')) = m'$$

olduğundan e_k bir birebir fonksiyondur.

A ve B kişilerinin C kişisinin de kullanılan şifreleme yöntemini bildiğini varsaymaları A ve B kişileri için en güvenli yoldur. Matematiksel olarak, C kişisinin de e ve d fonksiyonlarını bildiği fakat A ve B kişilerinin kullandığı k özel anahtarını bilmediği anlamına gelir. Örneğin, A ve B kişileri basit bir yer değiştirme şifreleme tekniği kullanıyorlarsa C kişisinin de bu gerçeğin farkında olduğunu varsaymaları gerekir. Dolayısıyla bir şifreleme sisteminin güvenliği, kullanılan şifreleme algoritmasının gizliliğinden daha çok kullanılan anahtarın gizliliğine bağlıdır. Bu anlayış modern kriptolojide *Kerckoff Prensibi* olarak isimlendirilir.

Eğer (K, M, C, e, d) işleyen, güvenilir bir şifre ise bu şifre aşağıdaki özelliklere sahip olmalıdır:

1. Her $k \in K$ anahtarı ve her $m \in M$ düz metni için $e_k(m)$ şifreli metnini hesaplamak kolay olmalıdır.

2. Her $k \in K$ anahtarı ve her $c \in C$ şifreli metni için $d_k(c)$ düz metnini hesaplamak kolay olmalıdır.

3. $k \in K$ anahtarı kullanılarak şifrelenmiş olan bir veya daha fazla $c_1, c_2, \ldots, c_n \in C$ şifreli mesajları verildiğinde bunlara karşılık gelen $d_k(c_1), d_k(c_2), \ldots, d_k(c_n)$ düz metinlerinin k anahtar bilgisi olmadan hesaplanması çok zor olmalıdır.

4. Bir veya daha fazla $(m_1, c_1), (m_2, c_2), \ldots, (m_n, c_n)$ düz metin ve bunlara karşılık gelen şifreli metin çiftleri için bu listede bulunmayan herhangi bir c şifreli metnin k anahtarı bilinmeden çözülmesi zor olmalıdır. Bu bilinen bir düz metin saldırısına karşı güvenlik özelliğidir.

5. C kişisi tarafından seçilmiş olan düz metinlerin herhangi $m_1, m_2, \ldots, m_n \in M$ listesi için karşılık gelen $e_k(m_1), e_k(m_2), \ldots, e_k(m_n)$ şifreli metinleri bilinse bile k anahtarı bilinmeksizin herhangi bir c şifreli metnini çözümlemek çok zor olmalıdır. Bu seçilmiş bir düz metin saldırısına karşı güvenlik özelliğidir.

1.3.1.Örnek. Basit bir yer değiştirme şifreleme tekniği yukarıda belirtilen 4.özelliğe sahip değildir. Çünkü bir tek (m, c) düz metin/şifreli metin çifti bile şifreleme tekniğinin çoğunu ortaya çıkarır. Bu nedenle basit bir yer değiştirme şifreleme tekniği bilinen düz metin saldırılarına karşı oldukça savunmasız bir şifreleme tekniğidir.

Kodlama Şemaları

Bir şifreleme sisteminde anahtarları, düz metinleri, şifreli metinleri birer sayı olarak ifade etmek ve bu sayıları da ikilik sistemde yazmak uygundur. Örneğin, bir *bit* 0 veya 1 olmak üzere, 0 dan 255 e kadar verilen sayılar 8 bitlik dizgiler halinde

$$a = 00000000,\ b = 00000001,\ c = 00000010, \ldots,\ z = 00011001$$

biçiminde alfabetik harflerin gösterilmesi için kullanılabilir. Küçük harfleri büyük harflerden ayırmak için A = 00011011, B = 00011100, … olarak alınabilir. Böylece bu kodlama yöntemi 256 tane farklı sembolün ikilik sisteme çevrilmesine olanak tanır.

Birçok bilgisayar verileri depolamak için yukarıdakine benzeyen ve adına ASCII kodlama denen bir kodlama sistemi kullanır. Bu kodlama sisteminin bir kısmı aşağıda görülmektedir. Örneğin, "Bed bug." ifadesi boşluk ve noktalama dahil olmak üzere ASCII koduna göre

B	**e**	**d**		**b**	**u**	**g**	**.**
66	101	100	32	98	117	103	46
01000010	01100101	01100100	00100000	01100010	01110101	01100111	00101110

biçiminde şifrelenir ve böylece bilgisayar tarafından

01000010011001010110010000100000011000100111010101100111001011 10

biçimindeki bir bit dizgisi biçiminde görür.

1.3.2. Tanım. Bir veri türünü başka bir veri türüne dönüştürme yöntemine *kodlama şeması* denir.

Örneğin, bir metni sayılara dönüştürmek bir kodlama şeması ile gerçeklenir. Kodlama şeması ile şifreleme şemalarının birbirlerinden farklı olmaları gerekir. Bir kodlama şemasının tamamen kamuya açık olduğu ve herkes tarafından aynı amaçlar için kullanıldığı varsayılır. Şifreleme şemaları ise gizli anahtara sahip olmayan herhangi birinden bilgi gizlemek için tasarlanırlar. Bu nedenle bir kodlama şeması bir şifreleme şemasında olduğu gibi bir kodlama fonksiyonu ve bu fonksiyonun tersi olan kodlamayı çözme (decoding) fonksiyonundan oluşur. Bununla birlikte bir kodlama şeması için her iki fonksiyon da kamuya açıktır ve hesaplaması hızlı ve kolaydır.

Bir kodlama şeması kullanılarak bir düz metin veya bir şifreli metin her bir blok sekiz bit olmak üzere sekizli bloklar halinde ifade edilebilir. Sekiz bitlik bir bloğa bir *byte* adı verilir, yani bir byte, 0 ile 255 arasında onluk sistemdeki bir sayı ile veya 00 ile FF arasında onaltılık sistemdeki ikililer ile ifade edilir. Bilgisayarlar genellikle bir zamanda bir seferde birden fazla byte işlem yapar. Örneğin, 64 bitlik bir işlemci bir zamanda her seferinde sekiz byte işlem yapar.

Kodlanmış Blokların Simetrik Şifrelenmesi

Daha önce de belirtildiği gibi bir kodlama şeması kullanılırken M düz metin mesajlarının uzayının sabit bir B uzunluğundaki dizgilerden, yani tam olarak B tane 0 ve 1 rakamlarından oluşan dizgilerden oluşur. Buradaki B sayısına şifrenin *blok boyutu* denir. Böylece genel bir düz metin mesajı, M uzayından seçilen mesaj bloklarının bir listesinden ibarettir ve şifreleme fonksiyonu, bu mesaj bloklarını her bloğu B bitlerin bir dizisi olan C uzayındaki bir şifreli metin blokları listesine dönüştürür. Eğer bu düz metin mesajı B bitten daha az bir blokla bitiyorsa bloğun sonu sıfırlarla biter. Orijinal düz metin mesajını M kümesinde bitlerin bloklarının bir dizisine dönüştüren bu kodlama işlemi kamuya açıktır.

Şifreleme ve şifre çözme her seferinde bir blok için yapıldığından bu işlemi bir tek düz metin mesajı, yani bir tek $m \in M$ için çalışmak yeterlidir. Dolayısıyla bir mesajı bloklara bölmek oldukça kullanışlıdır. Bir mesaj keyfi uzunlukta olabildiğinden şifreleme işlemini bir tek sabit uzunlukta parça üzerinde yapmak kolaylık sağlar. m düz mesaj metin bloğu bir B bitlerin dizgisidir ve bu dizgi ikili formda bir sayı ile özdeşleştirilir. Diğer bir deyişle, M uzayı, $0 \le m < 2^B$ özelliğindeki m tamsayılarının kümesi ile

$$\underbrace{m_{B-1}\, m_{B-2} \ldots m_2\, m_1\, m_0}_{m \text{ sayısının } B \text{ bitlerinin listesi}} \leftrightarrow \underbrace{m_{B-1} \cdot 2^{B-1} + m_{B-2} \cdot 2^{B-2} + \ldots + m_2 \cdot 2^2 + m_1 \cdot 2^1 + m_0 \cdot 2^0}_{0 \text{ ve } 2^{B-1} \text{ arasındaki tamsayılar}}$$

biçiminde özdeşleştirilir. Burada $m_0, m_1, \ldots, m_{B-1}$ sayılarının her biri 0 veya 1 dir.

Benzer şekilde, K anahtar uzayı ve C şifreli metin uzayı, belirli bir blok uzunluğundaki bit dizgilerinin tamsayı kümeleri ile özdeşleştirilir. Eğer anahtarlar, düz metin mesajları ve şifreli metin mesajları sırasıyla B_k, B_m ve B_c olarak gösterilirse K, M, C uzayları

$$K = \{k \in \mathbb{Z} \mid 0 \leq k < 2^{B_k}\}$$

$$M = \{m \in \mathbb{Z} \mid 0 \leq m < 2^{B_m}\}$$

$$C = \{c \in \mathbb{Z} \mid 0 \leq c < 2^{B_c}\}$$

biçiminde pozitif tamsayıların kümesi ile özdeşleştirilir. Böylece çok önemli bir soru ortaya çıkar: "A ve B kişileri K kümesini ne kadar büyük yapmalıdır?" veya denk olarak, "B_k anahtar bloğunu ne kadar büyük seçmelidir?" Eğer B_k çok küçükse, bir C kişisi A ve B kişilerinin gizli anahtarını bulana kadar 0 ile $2^{B_k} - 1$ arasındaki her sayıyı kontrol edebilir. Daha kesin olarak, C kişisinin d şifre çözme algoritmasını bildiği kabul edildiğinden C kişisi her bir $k \in K$ anahtarını alarak $d_k(c)$ yi hesaplamak için kullanır. C kişisinin geçerli ve geçersiz düz metin mesajları arasında ayrım yapabildiği varsayılırsa C kişisi bu işlemin sonucunda mesajı ele geçirir.

C kişisi anahtar uzayında geniş kapsamlı bir arama yaptığından bu saldırı *kaba kuvvet saldırısı* olarak adlandırılır. Diğer yandan şimdiki teknoloji ile anahtar uzayının en az 2^{80} elemana sahip olduğunda kapsamlı bir araştırmanın olanaksızdır. Böylece A ve B kişileri B_k değerini 80 sayısından büyük seçmelidirler.

Simetrik Şifrelerin Örnekleri

p büyük bir asal sayı, yani $2^{159} < p < 2^{160}$ olmak üzere A ve B kişilerinin K anahtar uzayını, M düz metin mesaj uzayını ve C şifreli metin mesaj uzayını $\mathbb{F}_p$ sonlu cismindeki birimlerin kümesini, yani

$$K = M = C = \{1, 2, \ldots, p-1\} = \mathbb{F}_p^*$$

olarak aldıklarını varsayalım.

A ve B kişileri rastgele bir $k \in K$ anahtarı seçerler, yani $1 \leq k < p$ özelliğinde bir k tamsayısı seçerler ve

$$e_k(m) \equiv k \cdot m \pmod{p} \tag{1.1}$$

biçiminde tanımlana e_k şifreleme fonksiyonunu kullanmaya karar verirler. Buna karşılık gelen d_k şifre çözme fonksiyonu, k', p modülüne göre k sayısının tersi olmak üzere

$$d_k(c) \equiv k' \cdot c \pmod{p}$$

dir. Dikkat edilirse, p çok büyük bir asal sayı olduğu halde genişletilmiş Öklid Algoritması ile k' sayısı, $2\log_2 p$ adımdan daha az adımda hesaplanabilir. Böylece k değerinden k' değerini elde etmek kriptolojide "kolay" sayılır.

Seçilebilecek yaklaşık 2^{160} olasılık var olduğundan bir C kişisinin k gizli anahtarını tahmin etmesi zordur. Üstelik C kişisi c şifreli metnini bilse bile k gizli anahtarını ele geçirmesi zordur. Dikkat edilirse

$$e_k : M \rightarrow C$$

şifreleme fonksiyonu herhangi bir k anahtarının seçimi için örtendir. Dolayısıyla her $c \in C$ ve herhangi bir $k \in K$ için $e_k(m) = c$ olacak şekilde bir $m \in M$ vardır. Ayrıca verilen herhangi bir şifreli metin, uygun bir anahtarla şifrelenmesi şartıyla herhangi bir düz metni temsil edebilir. Matematiksel olarak, bu herhangi bir $c \in C$ şifreli metin ve herhangi bir $m \in M$ düz metin verildiğinde $e_k(m) = c$ olacak şekilde bir k anahtarının var olduğunu ifade eder. Özellikle bu

$$k \equiv m^{-1} \cdot c \pmod{p} \tag{1.2}$$

anahtarı için de geçerlidir. Bu, A ve B kişilerinin şifrelerinin daha önce bahsedilen 1., 2. ve 3. özelliklerine sahip olduğunu gösterir. Gerçekten de, k anahtarını bilen herkes şifreleme ve şifre çözme işlemlerini kolayca yapabilir. Ancak k değerini bilmiyorsa şifre çözmek zordur. Bununla birlikte, bir tek bir düz metin/şifreli metin çifti (m,c) bile bir C kişisinin (1.2) denkliğini kullanarak k özel anahtarını bulmasına olanak sağladığından bu şifre 4. özelliğe sahip değildir.

Ayrıca A ve B kişileri şifreleme fonksiyonlarını $e_k(m) = k \cdot m$ olarak tanımlarsa bu şifre, yine 1. ve 2. özelliği gerçeklerken 3. özelliği gerçeklemez. Eğer C kişisi tek bir $c = k \cdot m$ şifreli metnini çözmeyi denerse bile bir büyük sayının çarpanlarına ayırma işlemi ile karşıya kalır. Bununla birlikte C kişisi c_1, c_2, …, c_n gibi birkaç şifreli metin elde edebilirse

$$\text{obeb}(c_1, c_2, \ldots, c_n) = \text{obeb}\,(k \cdot m_1, k \cdot m_2, \ldots, k \cdot m_n) = k \cdot (m_1, m_2, \ldots, m_n)$$

olarak yazılabilir. Dikkat edilirse en büyük ortak bölen bulma işlemi kolaydır.

Yukarıdaki örneğe göre p modülüne göre indirgeme işleminin bölünebilirlik gibi özellikleri yok eden bir "karıştırma" etkisi vardır. Bununla birlikte (1.1) şifresi bilinen bir düz metin saldırısına karşı savunmasız olduğundan sadece p modülüne göre indirgeme işlemi yapmak yeterli değildir. Eğer C kişisi hem şifreli metin hem de buna karşılık gelen m düz metnini elde ederse

$$k \equiv m^{-1} \cdot c \pmod{p}$$

değerini hesaplayarak anahtara kolayca ulaşabilir. Böylece tek bir düz metin/şifreli metin çifti bile anahtarı bulmak için yeterlidir. Dolayısıyla (1.2) şifreleme fonksiyonu da 4. özelliğe sahip değildir.

Yukarıda ele alınmış olan "çarpma-p modülüne göre indirgeme" şifresinin değişik versiyonları vardır. Örneğin,

$$e_k(m) \equiv m + k \pmod{p} \quad \text{ve} \quad d_k(c) \equiv c - k \pmod{p}$$

biçiminde verilen "toplama-p modülüne göre indirgeme" şifresi bunlardan birisidir. Bir başka versiyon, değiştirme şifresi ve çarpma şifresinin bileşimiyle oluşan *afin şifre*dir. denir. Bir afin şifresi için bir k anahtarı $k = (k_1, k_2)$ biçiminde iki tamsayıdan oluşur ve k_1', p modülüne göre k_1 değerinin tersi olmak üzere şifreleme ve şifreyi çözme fonksiyonları

$e_k(m) = k_1 \cdot m + k_2 \pmod{p}$

$$d_k(c) = k_1' \cdot (c - k_2) \pmod{p} \tag{1.3}$$

olarak tanımlanır.

Afin şifresinin bir genellemesi m düz metni, c şifreli metni ve k anahtarının ikinci kısmı olan k_2 değeri p modülüne göre n sayılarından oluşan sütun vektörleri ile değiştirilerek elde edilir. Bu şifreye *Hill şifresi* denir. Burada k anahtarının birinci kısmı olan k_1 değeri girdileri p modülüne göre tamsayılar olan $n \times n$ tipinde bir matris olarak alınır. Bu durumda şifreleme ve şifre çözme fonksiyonları yine (1.3)' te verildiği gibidir, ancak $k_1 \cdot m$ çarpımı bir matris ve bir vektörün çarpımıdır ve k_1', p modülüne göre k_1'in ters matrisidir. Afin ve Hill şifreleri de bilinen düz metin saldırılarına karşı savunmasızdır.

Asimetrik Şifreler

Daha önce görüldüğü gibi A ve B kişileri bir simetrik şifre kullanarak mesaj alışverişinde bulunmak isterlerse bir k gizli anahtarı belirlerler. Eğer A ve B kişileri gizli bir kanaldan haberleşirlerse bu şifreleme sistemi iyi olabilir. Ancak bir C kişisi bu kişilerin iletişiminden haberdar ise bu şartlar altında bir gizli anahtar alışverişi yapmak mümkün olabilir mi? Bu sorunun cevabı ilk bakışta hayır olarak düşünüldüğü halde Diffie ve Hellman (1976) bazı koşullar altında bunun mümkün olabileceğini belirtmişlerdir. Bu soruna etkili çözümler aramaya *açık anahtar* (veya *asimetrik*) *kriptolojisi* denir.

A kişisinin üst tarafında dar bir yuvası bulunan bir kasası olduğunu ve bu kasayı kamuya açık bir yere koyduğunu varsayalım. Üstelik bu kasanın oldukça güvenilir olduğu herkes tarafından bilinsin. B kişisinin A kişisine bir mesaj yazdığını ve bunu kasanın üst tarafındaki yuvadan içeri attığını düşünelim. Bu durumda sadece kasanın anahtarına sahip olan kişi, yani sadece A kişisi bu mesajı alır ve okur. Bu senaryoda, A kişisinin açık anahtarı kasadır ve şifreleme algoritması mesajı yuvaya koymak, şifreyi çözme algoritması ise anahtarla kasayı açmaktır. Dikkat edilirse bu yöntem gerçek

dünyada kullanılmaktadır. Örneğin, bir bankanın bankamatiğinde kullanılan para yatırma yuvaları bu biçimdedir. Bu para yatırma yuvaları güvenli olmalı, bir kişinin yuvayı kullanarak başka insanların mevduatlarına ulaşması mümkün olmamalıdır.

Yukarıda söz edilen "*yuvalı kasa*" kriptosistemi, kasa kamuya açık bir yerde bulunduğundan ve herkesin bu kasayı kullanarak *A* kişisine şifreli mesajlar gönderebildiğinden oldukça kullanışlıdır. Üstelik *A* kişisinin iletişime geçtiği her bir kişi için ayrı bir kasa oluşturması gerekmemektedir. Bundan başka *A* kişisinin kasayı açarak başka kişiler *A* kişisine mesaj göndermeden önce *B* kişisinin gönderdiği mesajı silmesi gerekmemektedir.

Şimdi bunu matematiksel olarak açıklayalım. Daha önce olduğu gibi, anahtarların bulunduğu bir *K* uzayı, düz mesajların bulunduğu bir *M* uzayı ve şifreli mesajların bulunduğu bir *C* uzayı vardır. Bununla birlikte bir $k \in K$ anahtarı

$$k = (k_{\text{öz}}, k_{\text{aç}})$$

biçiminde adlarına sırasıyla *özel anahtar* (*private key*) ve *açık anahtar* (*public key*) denilen bir anahtar çiftinden oluşur. Her bir $k_{\text{aç}}$ açık anahtarı için

$$e_{k_{aç}} : M \to C$$

biçiminde verilen bir şifreleme fonksiyonu ve her bir $k_{\text{öz}}$ özel anahtarı için

$$d_{k_{öz}} : C \to M$$

biçiminde verilen bir şifre çözme fonksiyonu vardır. Eğer $k = (k_{\text{öz}}, k_{\text{aç}}) \in K$ ise her $m \in M$ için $d_{k_{öz}}(e_{k_{aç}}(m)) = m$ dir.

Bir asimetrik şifre güvenli ise bir *C* kişisi $k_{\text{aç}}$ açık anahtarını bilse bile $d_{k_{öz}}$ şifre çözme fonksiyonunu hesaplayamaz. Bu durumda *A* kişisi bir *B* kişisine güvenli olmayan bir iletişim kanalından $k_{\text{aç}}$ anahtarını gönderebilir ve *B* kişisi, *A* kişisine $(e_{k_{aç}}(m))$ şifreli mesajını gönderir ve bir başka *C* kişisi bu mesajı çözemez. Bu şifreli mesajın şifresinin

kolayca çözümü $k_{\text{öz}}$ anahtarını kullanmakla mümkündür ki bu anahtarı sadece *A* kişisi bilmektedir. $e_{k_{aç}}$ fonksiyonunun tersini hesaplamak için *A* kişisinin kullandığı $k_{\text{öz}}$ özel anahtarına, *kestirme bilgisi* (*trapdoor information*) denir. $k_{\text{öz}}$ ve $k_{\text{aç}}$ anahtarlarının birbirinden farklı olması şifreyi asimetrik yapar.

2. KURAMSAL TEMELLER ve KAYNAK ARAŞTIRMASI

Ayrık logaritma problemi, şifreleme, anahtar değişimi, dijital imazalar ve belli bir çıktıya sahip fonksiyonlar (hash fonksiyonları) gibi birçok kriptografik yapıda kullanılır. Ayrık logaritma problemine dayanan ilk açık anahtar kriptosistemi Diffie ve Hellman (1976) tarafından "*New directions in cryptography*" isimli makalede ele alınmıştır. Daha sonraki yıllarda Rivest, Shamir ve Adleman (1978) tarafından güvenilirliği büyük sayıların çarpanlamasına bağlı olan Rivest, Shamir, Adleman kriptosistemleri (RSA kriptosistemleri) ele alınmıştır. Bununla birlikte Diffie ve Hellman, güvenilirliği $\mathbb{F}_p^*$ grubundaki ayrık logaritma probleminin çözümüne dayanan bir anahtar değişim algoritması vermişlerdir. Daha sonra, ElGamal (1985) ayrık logaritma problemine dayanan bir açık anahtar kriptosistemi oluşturulmuştur. Koblitz (1987) ve Miller (1986), ayrık logaritma probleminin çözümünün daha zor olabileceğini düşünerek, $\mathbb{F}_p^*$ grubu yerine $\mathbb{F}_p$ sonlu cismi üzerinde tanımlı eliptik eğri üzerindeki noktaların grubu olan $E(\mathbb{F}_p)$ grubunu almışlar ve bu düşünce ışığında eliptik eğri kriptolojisi ortaya çıkmıştır.

Bu çalışmada özellikle açık anahtar kriptosistemleri ile ilgili olan ayrık logaritma problemi ele alınacaktır.

2.1. Ayrık Logaritma Problemi

Bu kısımda ayrık logaritma problemi tanıtılacak ve bu problemin çözümü için geliştirilmiş çeşitli algoritmalar üzerinde durulacaktır.

2.1.1. Tanım. G bir grup, $x, y \in G$ olmak üzere y, x ile üretilen alt grubun bir elemanı olsun.

$$x^m = y$$

olacak biçimdeki $m \geq 1$ tamsayısının bulunması problemine G grubu için *ayrık logaritma problemi* (ALP) denir. $x^m = y$ eşitliğini gerçekleyen en küçük m tamsayısına y elemanının x elemanına göre *logaritması* (*indeksi*) denir ve $m = \log_x(y)$ veya $m = ind_x(y)$ ile gösterilir.

2.1.2. Uyarı 1. Ayrık logaritma problemi bir G grubu için tanımlandığından bu problemin çözümü bazı gruplar için kolay bazı gruplar için zordur. Örneğin, ayrık logaritma probleminin $(\mathbb{F}_p, +)$ grubundaki çözümü oldukça kolaydır. Gerçekten de $(\mathbb{F}_p, +)$ grubu için ayrık logaritma problemi, $x, y \in \mathbb{F}_p$ olmak üzere $mx = y$ olacak biçimdeki m tamsayısının bulunmasıdır ve bu denklemin çözümü için x elemanının $\mathbb{F}_p$ cismindeki tersinin belirlenmesi gerekir. $x \in \mathbb{F}_p$ nin tersinin belirlenmesi ise Euclid algoritması kullanılarak $O(\log p)$ zaman alır. Benzer biçimde $(\mathbb{Z}_m, +)$, $(\mathbb{R}^*, \cdot)$, $(\mathbb{C}^{\mathbb{I}}, \cdot)$ grupları için de bu problem oldukça kolaydır.

2. Bununla birlikte, ALPnin çözümü $(\mathbb{F}_p^*, \cdot)$ grubu için zordur: $(\mathbb{F}_p^*, \cdot)$ grubu için ayrık logaritma problemi $x, y \in \mathbb{F}_p^*$ olmak üzere $x^m = y$ olacak biçimdeki m tamsayısının belirlenmesidir. Daha sonra görüleceği gibi, $O(p)$ mertebeli bir grupta ayrık logaritma problemi $O(\sqrt{p})$ adımda çözülebilir. Ancak ayrık logaritma problemini $\mathbb{F}_p^*$ grubunda çözmek için daha hızlı algoritmalar da bulunabilir. $\mathbb{F}_p^*$ grubunda ALPni çözmek için bilinen en iyi algoritma *indeks hesabı* yöntemidir ve bu algoritma ALPni c belli bir sabit olmak üzere

$$\exp(c\sqrt[3]{(\log p)(\log\log p)^2})$$

zamanda çözer. Bu zaman üstel zamandan daha hızlı fakat polinom zamandan da daha yavaş olduğundan *altüstel zaman* olarak adlandırılır.

3. Yukarıda da görüldüğü gibi, ayrık logaritma probleminin çözümünde çeşitli algoritmalar kullanılır. Kullanım açısından, ayrık logaritma probleminin üstel bir

zamanda çözülebileceği bir G grubu alınır. Bu çalışmada ayrık logaritma problemi özellikle $\mathbb{F}_p^*$ sonlu grubu ve sonlu bir cisim üzerinde tanımlı bir eliptik eğrinin üzerindeki noktaların oluşturduğu $E(\mathbb{F}_p)$ grubu için ele alınacaktır.

4. Bir G grubu için verilmiş olan ayrık logaritma problemi, grubun bir sonlu grup olması durumunda aşağıdaki hali alır.

2.1.3. Tanım. $g \in \mathbb{F}_p$ bir ilkel kök olmak üzere h, $\mathbb{F}_p$ cisminin sıfırdan farklı elemanı olsun.

$$g^m \equiv h \pmod{p}$$

olacak biçimdeki $m \geq 1$ tamsayısının bulunması problemine $\mathbb{F}_p$ *sonlu cismi üzerinde ayrık logaritma problemi* denir. Bu özellikteki m tamsayısına g tabanına göre h elemanının *ayrık logaritması* denir ve $\log_g(h)$ ile gösterilir.

2.1.4. Uyarı 1. $g^m \equiv h \pmod{p}$ denkliğini gerçekleyen bir m çözümü var ise sonsuz tane çözüm bulunabilir. Buna göre m, bu denkliğin bir çözümü ise her $k \in \mathbb{Z}$ için $m + k(p - 1)$ de bir çözümdür. Gerçektende, Fermat'nın Küçük Teoremi gereği, $g^{p-1} \equiv 1 \pmod{p}$ olduğundan

$$g^{m+k(p-1)} = g^m(g^{p-1})^k \equiv h \cdot 1^k \equiv h \pmod{p}$$

dir, yani $\log_g(h)$ sayısı $p - 1$ modülüne göre tanımlanır. Üstelik

$$\log_g : \mathbb{F}_p^{\mathbb{I}} \rightarrow \mathbb{Z}_{p-1}$$

fonksiyonunun iyi tanımlı olduğu kolayca görülebilir.

2. Her $a, b \in \mathbb{F}_p^*$ için

$$\log_g(ab) = \log_g(a) + \log_g(b)$$

yani

$$\log_g(ab) \equiv \log_g(a) + \log_g(b) \pmod{p-1}$$

dir. O halde $\log_g$ fonksiyonunda klasik logaritma fonksiyonunda olduğu gibi çarpma işlemi toplamaya dönüşür.

2.1.5. Örnek 1. $p = 56509$ ve $g = 2$ olmak üzere

$$2^m \equiv 38679 \pmod{56509}$$

olacak biçimdeki m tamsayısını belirlemek için 2 elemanının

$$2^2, 2^3, 2^4, \ldots \pmod{56509}$$

olacak biçimde tüm kuvvetleri belirlenebilir. Ancak bu yol oldukça zordur. Bununla birlikte bir bilgisayar kullanılarak $m = 11235$ olduğu görülebilir (J. Hoffstein ve ark. 2008). Aşağıda böyle bir m sayısının belirlenmesi için indeks hesabı yöntemi kullanılmıştır. Bu yöntem Kısım 2.7 de detaylı olarak ele alınacaktır.

2. $p = 1217$ ve $g = 3$ olmak üzere $3^m \equiv 37 \pmod{1217}$ olacak biçimdeki m sayısını belirlemek için küçük asal sayılardan oluşan adına *çarpım tabanı* denilen

$$B = \{2, 3, 5, 7, 11, 13\}$$

kümesini dikkate alalım. İlk olarak

$$3^x \equiv \pm B \text{ kümesindeki belli asal sayıların çarpımı} \pmod{1217}$$

olacak biçiminde x sayılarını belirleyelim. Bu sayılardan bazıları;

$$3^1 \equiv 3 \pmod{1217}$$

$$3^{24} \equiv -2^2 \cdot 7 \cdot 13 \pmod{1217}$$

$$3^{25} \equiv 5^3 \pmod{1217}$$

$$3^{30} \equiv -2 \cdot 5^2 \pmod{1217}$$

$$3^{54} \equiv -5 \cdot 11 \pmod{1217}$$

$$3^{87} \equiv 13 \pmod{1217}$$

biçimindedir. Diğer yandan $3^{(p-1)/2} \equiv -1 \pmod{p}$ olduğundan $\log_3(-1) = 608$'dir. Dolayısıyla

$$1 \equiv \log_3(3) \pmod{1216}$$
$$24 \equiv 608 + 2\log_3(2) + \log_3(7) + \log_3(13) \pmod{1216}$$
$$25 \equiv 3\log_3(5) \pmod{1216}$$
$$30 \equiv 608 + \log_3(2) + 2\log_3(5) \pmod{1216}$$
$$54 \equiv 608 + \log_3(5) + \log_3(11) \pmod{1216}$$
$$87 \equiv \log_3(13) \pmod{1216}$$

olarak yazılabilir. Birinci denklikten $\log_3(3) \equiv 1$ dir. Diğer yandan üçüncü denklikten $\log_3(5) \equiv 819 \pmod{1216}$ ve altıncı denklikten $\log_3(13) \equiv 87$ olduğu elde edilir. Böylece dördüncü denklikten

$$\log_3(2) \equiv 30 - 608 - 2 \cdot 819 \equiv 216 \pmod{1216}$$

denkliği ve beşinci denklikten

$$\log_3(11) \equiv 54 - 608 - \log_3(5) \equiv 1059 \pmod{1216}$$

denkliği elde edilir. Son olarak ikinci denklikten

$$\log_3(7) \equiv 24 - 608 - 2\log_3(2) - \log_3(13) \equiv 113 \pmod{1216}$$

denkliği elde edilir. Yukarıdaki denklikler dikkate alındığında çarpım tabanındaki tüm elemanların ayrık logaritmalarının belirlenmiş olduğu görülür.

$3^m \equiv 37 \pmod{1216}$ olacak biçimdeki m sayısını bulmak için $3^k \cdot 37 \pmod{p}$ değeri B kümesindeki asal sayıların çarpımı olarak yazılabileceği bir k değeri seçilir. Böylece $k = 16$ için

$$3^{16} \cdot 37 \equiv 2^3 \cdot 7 \cdot 11 \pmod{1217}$$

olduğundan

$$\log_3(37) \equiv 3\log_3(2) + \log_3(7) + \log_3(11) - 16 \equiv 588 \pmod{1216}$$

denkliği elde edilir. O halde $m = 588$ dir (Washington 2003).

Burada B çarpım tabanı kümesinin büyüklüğünün seçimi önemlidir. Eğer B kümesi çok küçük alınırsa B kümesindeki asal sayıları kullanarak g nin kuvvetlerini bulmak zorlaşır. Eğer B kümesi çok büyük alınırsa bağlantıları bulmak kolay olduğu halde B kümesinin elemanlarının ayrık logaritmalarını belirlemek için lineer cebir kullanılır. Bu ise oldukça kullanışsızdır.

2.2. Diffie-Hellman Anahtar Değişimi

Bu kısımda kriptogarfik anahtarların değişimi işleminde kullanılan ve ayrık logaritma problemine dayanan Diffie-Hellman anahtar değişimi ele alınacaktır. Diffie-Hellman anahtar değişimi sisteminde X ve Y kişileri simetrik bir şifreleme yöntemi kullanarak veri alışverişi yapmak için ortak bir anahtar belirlerler. Örneğin, X ve Y kişileri finansal verileri iletmek isteyen bankalar olabilir. Bu ortak anahtarın teslimi için bir kurye kullanmak güvenilir ve pratik değildir. Dahası X ve Y kişilerinin daha önce iletişime geçmediği ve bu nedenle aralarındaki tek iletişim kanalının kamuya açık olduğu varsayılabilir. Diffie ve Hellman gizli bir anahtar oluşturmak için aşağıdaki biçimde hareket edilebileceğini söylemişlerdir. Buna göre,

- X ve Y kişileri büyük bir p asal sayısı ve p modülüne göre sıfırdan farklı bir g tamsayısı belirlerler. Daha sonra bu kişiler, p asal sayısını ve $\mathbb{F}_p^*$ grubundaki mertebesi büyük bir asal sayı olacak biçimde seçilen bir g değerini kamuya açık bir kanalda paylaşırlar. Örneğin, bu değerleri kendi internet sitelerinde yayınlayabilirler ve böylece bu değerlerden üçüncü bir Z kişisi de haberdar olur.

- X kişisi gizli bir a tamsayısı seçer ve $A \equiv g^a \pmod{p}$ değerini hesaplar. Y kişisi de gizli bir b tamsayısı seçer ve $B \equiv g^b \pmod{p}$ değerini hesaplar.

- X kişisi A değerini Y kişisine gönderir ve Y kişisi de B değerini X kişisine gönderir. X ve Y kişileri güvenli olmayan iletişim kanalı üzerinden bu değerleri gönderdiğinden Z kişisi de A ve B değerlerinden haberdardır.

- X kişisi $A' \equiv B^a \pmod{p}$ değerini ve Y kişisi de $B' \equiv A^b \pmod{p}$ değerini hesaplar. A' ve B' değerleri aslında aynıdır. Gerçekten de,

$$A' \equiv B^a \equiv (g^b)^a \equiv g^{ab} \equiv (g^a)^b \equiv A^b \equiv B' \pmod{p}$$

  dir.

Böylece X ve Y kişileri Diffie-Hellman anahtar değişimi sistemi ile veri alışverişinde bulunurlar.

2.2.1. Örnek. X ve Y kişilerinin $p = 953$ asal sayısını ve $g = 629 \in \mathbb{F}^{\mathbb{I}}_{953}$ ilkel kökünü kullandıklarını varsayalım. X kişisi $a = 381$ gizli anahtarını seçer ve $A = 196 \equiv 629^{381} \pmod{953}$ değerini hesaplar. Benzer şekilde, Y kişisi $b = 781$ gizli anahtarını seçer ve $B = 289 \equiv 629^{781} \pmod{953}$ değerini hesaplar. X kişisi Y kişisine 196 sayısını ve Y kişisi de X kişisine 289 sayısını gönderir. Bu iletişim güvensiz bir kanal üzerinden yapıldığından halka açıktır ve $A = 196$ ile $B = 289$ değerleri gizli kalamaz. Burada sadece $a = 381$ ve $b = 781$ sayıları gizlidir. X ve Y kişileri

$$A' = 289^{381} \equiv 839 \pmod{953} \quad \text{ve} \quad B' = 196^{781} \equiv 839 \pmod{953}$$

değerlerini hesaplar. Buradaki ortak değer, yani 839 sayısı onların gizli değeridir.

Eğer Z kişisinin bu anahtar değişimini gördüğü varsayılır ve Z kişisi

$$629^a \equiv 196 \pmod{953} \qquad \text{ve} \qquad 629^b \equiv 289 \pmod{953}$$

denkliklerinin her ikisini de çözebilirse X ve Y kişilerinin gizli anahtarını ele geçirebilir. Z kişisinin X ve Y kişilerinin yardımı olmadan gizli anahtarı ele geçirebilmesinin tek yolu budur.

Örneğe dikkat edilirse, X ve Y kişilerinin hesapladığı A, B, $A' = B'$ değerleri oldukça küçük, yani hesaplaması kolaydır. Ayrıca Z kişisinin bilgisayarının da 953 modülüne göre 629 sayısının mümkün olan tüm kuvvetlerini kontrol etmesi çok az zaman alır. Bu ise bu bilgi aktarımının güvenli olmadığını gösterir. Son bilgisayar teknolojileri göz önüne alınırsa, X ve Y kişilerinin güvenli bir şekilde bilgi aktarımı yapabilmeleri için p asal sayısını yaklaşık olarak 1000 bit ve asal mertebeli g elemanını yaklaşık $p/2$ olarak seçmelerinin uygun olduğu düşünülebilir.

Z kişisi A ve B değerlerini bildiğinden g^a ve g^b değerlerini de bilmektedir. Ayrıca g ve p değerlerini de bildiğinden ayrık logaritma problemini çözebilir. Böylece a ve b değerlerini bulabilir. Bu şekilde X ve Y kişilerinin gizli değeri olan g^{ab} değerini hesaplayabilir. X ve Y kişileri, ancak güvenliği sağladıklarında Z kişisinin ayrık logaritma problemini çözmesi olanaksız hale gelir. Bu durumda X ve Y kişilerinin paylaştığı ortak değerin güvenliği, aşağıda tanımlanacak olan Diffie-Hellman problemine bağlıdır.

2.2.2. Tanım. p bir asal sayı ve g bir tamsayı olsun. $g^a \pmod p$ ve $g^b \pmod p$ bilinen değerlerinden $g^{ab} \pmod p$ değerini hesaplama problemine *Diffie-Hellman problemi* (DHP) denir.

Diffie-Hellman probleminin ayrık logaritma probleminden daha zor olmadığı açıktır. Eğer Z kişisi ayrık logaritma problemini çözebilirse, ele geçirilen

$$A \equiv g^a \text{ ve } B \equiv g^b$$

değerlerinden X ve Y kişilerinin gizli anahtarları olan a ve b değerlerini hesaplayabilir. Doğal olarak, Z kişisi için a ve b değerlerini bulduktan sonra g^{ab} değerini hesaplamak kolay olacaktır.

2.3. ElGamal Açık Anahtar Kriptosistemi

Açık anahtar kriptosistemleri ilk olarak Diffie-Hellman tarafından ele alındığı halde Taher ElGamal'ın 1985 yılında tanımladığı ElGamal açık anahtar kriptosistemi ile birlikte gelişmiştir. Bu sistem ayrık logaritma problemine dayalıdır ve Diffie-Hellman anahtar alışverişi ile ilgili bir asimetrik şifreleme algoritmasıdır. Bu kısımda $\mathbb{F}_p^*$ grubu için ayrık logaritma problemine dayalı olarak ElGamal açık anahtar kriptosistemi ele alınacaktır. Açık anahtar kriptosistemlerinin en önemlilerinden birisi olan ElGamal açık anahtar kriptosisteminde X kişisi bir açık anahtar ve bir algoritma yayınlar. Bu sistemde açık anahtar bir sayı, algoritma ise Y kişisinin X kişisinin açık anahtarını kullanarak kendi mesajını şifrelediği yöntemdir. X kişisinin kimseye açıklamadığı özel bir anahtarı vardır ve üstelik sadece X kişisi açık anahtarı kullanarak şifrelenmiş mesajların şifresini çözebilir.

ElGamal açık anahtar kriptosistemini kullanmak için X kişisi $\mathbb{F}_p^*$ grubu için ayrık logaritma probleminin çözümünün zor olduğu bir büyük p asal sayısı ve bir büyük asal mertebeli $g \in \mathbb{F}_p^*$ elemanını seçer. X kişisi p ve g değerlerini kendisi seçebilir veya p ve g değerleri güvenilir bir kaynak tarafından önceden seçilmiş de olabilir.

X kişisi özel anahtar olarak gizli bir a sayısı seçer ve

$$A \equiv g^a \pmod{p}$$

değerini hesaplar. ElGamal açık anahtar kriptosisteminde de Diffie-Hellman anahtar değişiminde olduğu gibi X kişisi özel anahtarı olan a sayısını gizli tutar ve açık anahtar olan A değerini yayınlar.

Y kişisinin, X kişisinin açık anahtarı olan A değerini kullanarak bir m mesajını şifrelemek istediğini ve üstelik m mesajının 2 ile p arasında bir tamsayı olduğunu varsayalım. Y kişisi m mesajını şifrelemek için p modülüne göre rastgele bir k sayısı

seçer. Y kişisinin seçmiş olduğu k sayısı sadece ve sadece bir mesajı şifrelemek için kullanıldığından k sayısına bir *geçici anahtar* denir.

Y kişisi, m mesajını, rastgele seçilen k geçici anahtarını ve X kişisinin açık anahtarı olan A değerlerini kullanarak

$$c_1 \equiv g^k \pmod{p} \qquad \text{ve} \qquad c_2 \equiv mA^k \pmod{p}$$

değerlerini hesaplar. Böylece Y kişisinin X kişisine göndermiş olduğu şifreli metin, yani m mesajının şifrelenmiş hali, (c_1, c_2) sayı çiftidir. X kişisinin Y kişisine göndermiş olduğu bu şifreli metni çözmesi için izlemesi gereken yol aşağıdaki şekildedir:

X kişisi a özel anahtarını bildiğinden

$$x \equiv {c_1}^a \pmod{p}$$

değerini ve böylece $x^{-1} \pmod{p}$ değerini hesaplar. Daha sonra X kişisi x^{-1} ile c_2 yi çarpar ve $x \equiv {c_1}^a \pmod{p}$ olduğunu kullanarak

$$x^{-1} \cdot c_2 \equiv ({c_1}^a)^{-1} \cdot c_2 \pmod{p}$$

denkliğini elde eder. Bu denklikte $c_1 \equiv g^k$, $c_2 \equiv mA^k \pmod{p}$ olduğunu göz önüne alarak

$$x^{-1} \cdot c_2 \equiv (g^{ak})^{-1} \cdot (mA^k) \pmod{p}$$

denkliğini elde eder. Daha sonra $A \equiv g^a \pmod{p}$ denkliğini kullanarak

$$x^{-1} \cdot c_2 \equiv (g^{ak})^{-1} \cdot (m(g^a)^k) \pmod{p}$$

denkliğini elde eder. Son olarak $g^{ak} \cdot (g^{ak})^{-1} = 1$ olduğundan bu son denklik

$$x^{-1} \cdot c_2 \equiv m \pmod{p}$$

halini alır. Böylece X kişisi, x^{-1} ile c_2 yi çarparak m mesajına ulaşmış olur.

X ve Y kişilerinden farklı bir Z kişisinin mesajın şifresini çözmek istediğini düşünelim. p ve g parametreleri ve açık anahtar olan $A \equiv g^a \pmod{p}$ değeri halka açık bir kanaldan yayınlandığından Z kişisi bu değerleri bilmektedir. Eğer Z kişisi ayrık logaritma problemini çözebilirse, a özel anahtarını bulur ve mesajın şifresini de çözer. Dolayısıyla bu mesajın şifresinin çözülmesi ayrık logaritma probleminin çözümüne bağlıdır.

2.3.1. Örnek. X kişisinin $p = 521$ asal sayısını ve $g = 6$ ilkel kökünü kullandığını varsayalım. X kişisi kendisine seçmiş olduğu $a = 165$ özel anahtarını kullanarak

$$A \equiv g^a \equiv 6^{165} \equiv 175 \pmod{521}$$

açık anahtarını hesaplar. Y kişisi X kişisine $m = 407$ mesajını göndermeye karar verir ve rasgele seçilmiş bir $k = 112$ anahtarını belirler. Y kişisi bilinen bu değerlerle birlikte

$$c_1 \equiv 6^{112} \equiv 59 \pmod{521} \quad \text{ve} \quad c_2 \equiv 407 \cdot 175^{112} \equiv 511 \pmod{521}$$

hesaplamalarını yapar. Böylece Y kişisi X kişisine göndermek istediği şifreli metni $(c_1, c_2) = (59, 511)$ sayı çifti olarak belirlemiş olur.

X kişisi Y kişisinin gönderdiği m mesajını çözmek için olan $a = 165$ özel anahtarını kullanarak

$$x \equiv {c_1}^a \equiv 59^{165} \equiv 320 \pmod{521} \text{ ve } x^{-1} \equiv 324 \pmod{521}$$

hesaplamalarını yapar. Böylece X kişisi

$$c_2 \, x^{-1} \equiv 511 \cdot 324 \equiv 407 \pmod{521}$$

hesaplamasını yaparak gönderilmiş olan m mesajını çözmüş olur.

2.3.2. Uyarı 1. ElGamal kriptosisteminde m düz metin mesajı 2 ile $p - 1$ arasında bir tamsayı iken şifrelenmiş metin mesajı 2 ile $p - 1$ arasındaki c_1 ve c_2 tamsayılarından oluşur. Böylece genellikle, şifreli mesajı yazmak için düz mesajı yazmak için gerekli olan bit sayısının iki katı kadar bit gereklidir. Dolayısıyla ElGamal, 2 ye 1 mesaj sistemidir.

Modern kriptosistemlerinin amaçlarından birisi, kriptosistemlerin güvenirliğinin bağlı olduğu bir zor problem belirlemektir. Örneğin, ElGamal kriptosisteminin güvenirliği, Diffie-Hellman probleminin çözümünün zorluğuna bağlıdır. Böylece aşağıdaki önermede görüleceği gibi, ElGamal kriptosistemi ile elde edilen şifreleri çözebilen kişilerin Diffie-Hellman problemini de çözebileceği sonucu elde edilir.

2.3.3. Önerme. p bir asal sayı ve $g \in \mathbb{F}_p^*$ olsun. Z kişisinin ElGamal açık anahtarlarını kullanarak şifrelenmiş keyfi ElGamal şifreli mesajlarını çözen bir sisteme (oracle) eriştiğini varsayalım. Bu durumda Z kişisi Diffie-Hellman problemini çözmek için bu sistemi kullanabilir (Hoffstein ve ark. 2008).

İspat. Diffie-Hellman problemini çözmek için ElGamal oracle sistemini nasıl kullanabileceğini açıklayalım. Diffie-Hellman probleminde Z kişisinin

$$A \equiv g^a \pmod{p} \text{ ve } B \equiv g^b \pmod{p}$$

değerlerini bildiği hatırlanırsa Z kişisinin $g^{ab} \pmod{p}$ değerini hesaplaması gereklidir. Burada Z kişisinin gizli a ve b değerlerini bilmediği unutulmamalıdır.

Z kişisinin bir ElGamal oracle sistemini kullandığını varsayalım. Z kişisi, bir p asal sayısını, bir g elemanını, belirli bir A açık anahtarını ve belirli bir (c_1, c_2) şifreli metnini sisteme gönderebilir. Bu sistem Z kişisine

$$(c_1{}^a)^{-1} \cdot c_2 \pmod{p}$$

değerini geri döndürür. Eğer Z kişisi Diffie-Hellman problemini çözmek istiyor ise $c_1 = B = g^b$ ve $c_2 = 1$ olarak seçebilir. Bu seçimle birlikte, sistem Z kişisine $(g^{ab})^{-1} \pmod{p}$ değerini verir, Z kişisi de bu değerin p modülüne göre tersini alarak ve $g^{ab} \pmod{p}$ değerini hesaplamış ve böylece Diffie-Hellman problemini çözmüş olur.

Z kişisi keyfi bir c_2 değeri seçer ve A açık anahtarı ile (B, c_2) şifreli metnini oracle sistemine gönderirse sistem Z kişisine asıl ulaşmak istediği m düz metin mesajının

$$m \equiv (c_1{}^a)^{-1} \cdot c_2 \equiv (B^a)^{-1} \cdot c_2 \equiv (g^{ab})^{-1} \cdot c_2 \pmod{p}$$

denkliğini gerçeklediğini belirtir. m değerini bilen Z kişisi $g^{ab} \pmod{p}$ değerini bulmak için

$$m^{-1} \cdot c_2 \equiv g^{ab} \pmod{p}$$

hesabını yapar. Böylece Z kişisi a ve b gizli değerlerini bilmeden, sistem yardımı ile $g^{ab} \pmod{p}$ değerini hesaplamış olur. Bundan başka, Z kişisi ayrık logaritma problemini çözmediği halde, yalnızca Diffie-Hellman problemini çözmüş olur.

Z kişisinin keyfi şifrelemeleri çözen bir oracle sistemine erişimi bir saldırıdır ve bu saldırı *seçilmiş şifreli metin saldırısı* olarak bilinir. O halde, bir Diffie-Hellman probleminin zor olması ElGamal sisteminin seçilmiş şifreli metin saldırılarına karşı güvenli olduğunu gösterir.

2.4. Ayrık Logaritma Problemi İçin Bir Çarpışma Algoritması

Bu kısımda bir çarpışma veya ortada buluşma algoritması örneği olan bir ayrık logaritma algoritması tanımlanacaktır. Daniel Shanks tarafından ele alınmış olan bu algoritma yalnızca $\mathbb{F}_p^*$ grubunda değil, herhangi bir grupta da çalışmaktadır.

2.4.1. Önerme. G bir grup ve $g \in G$ elemanının mertebesi N olsun. Bu durumda $g^x = h$ ayrık logaritma problemi, her adım g elemanı ile çarpma işleminden oluşmak üzere $O(N)$ adımda çözülür (Hoffstein ve ark. 2008).

İspat. $x = 0, 1, \ldots, N-1$ için g^x değerleri bulunur ve listelenir. Burada her ardışık değer g elemanının önceki değerle çarpılmasıyla elde edilir. $g^x = h$ eşitliğinin bir çözümü varsa h listede bulunur.

2.4.2. Uyarı. Eğer $G = \mathbb{F}_p^*$ olarak alınırsa, her bir $g^x \pmod p$ değerinin hesaplanması $O((\log p)^k)$ bilgisayar işlemi gerektirir, burada k ve büyük-O sabitleri, bilgisayara ve çarpma işlemi için kullanılan algoritmaya bağlıdır. Böylece bilgisayar adımlarının tüm sayısı ya da *çalışma süresi*, N, g elemanının mertebesi olmak üzere $O(N(\log p)^k)$ kadardır. Genel olarak, $O((\log p)^k)$ çarpanı ihmal edilerek, $O(N)$ zamanda çalışması tercih edilecektir.

Aşağıda verilecek olan çarpışma algoritmasının amacı, iki liste yaparak bu iki listedeki ortak olan elemanı bulmaktır. Ayrıca bu algoritmanın çalışma süresi $O(\sqrt{N})$ adımdan biraz daha fazladır.

2.4.3. Önerme (Shanks'in Bebek Adımı-Dev Adımı Algoritması). G bir grup ve $g \in G$ mertebesi $N \geq 2$ olan bir eleman olsun. Aşağıdaki algoritma $g^x = h$ ayrık logaritma problemini $O(\sqrt{N} \log N)$ adımda çözer.

1. $n = 1 + \lfloor \sqrt{N} \rfloor$ al, böylece $n > \sqrt{N}$ dir.

2. $e, g, g^2, g^3, \ldots, g^n$ elemanlarının listesini oluştur (buradaki g elemanı ile çarpma işlemleri bebek adımlarıdır).

3. $u = g^{-n}$ olmak üzere $h, h \cdot u, h \cdot u^2, h \cdot u^3, \ldots, h \cdot u^n$ elemanlarının listesini oluştur (buradaki u elemanı ile çarpma işlemleri dev adımlardır).

4. (2) ve (3) adımlarındaki listeler arasında bir eşleme bul. Eğer $g^i = h \cdot u^j$ ise $h = g^{i+jn}$ dir, aksi halde h, g nin bir kuvveti değildir (Hoffstein ve ark. 2008).

İspat. (2) ve (3) adımlarındaki iki listeyi oluşturmak için yaklaşık olarak $2n$ çarpma işlemi yapılmıştır. Şimdi bu listeler arasında bir eşleme olduğunu varsayalım. Standart sınıflandırma ve arama algoritmalarını kullanarak $\log(n)$ adımın bir küçük katında bir eşleme bulunabilir, dolayısıyla bu iki liste arasında eşleme bulmak $O(n \log n)$ adım alır. Böylece $n \approx \sqrt{N}$ olduğundan

$$n \log n \approx \sqrt{N} \log \sqrt{N} = \frac{1}{2} \sqrt{N} \log N$$

dir. O halde algoritmanın çalışma süresi $O(n \log n) = O(\sqrt{N} \log N)$ olur.

Şimdi (2) ve (3) adımlarında böyle bir eşlemenin her zaman var olduğunu görelim. Bunun için x, $g^x = h$ eşitliğinin bir çözümü olsun. Bu durumda $0 \le r < n$ olmak üzere $x = nq + r$ olarak yazılabilir. $1 \le x < N$ ve $n > \sqrt{N}$ olduğundan

$$q = \frac{x - r}{n} < \frac{N}{n} < n$$

eşitsizliği elde edilir. O halde $g^x = h$ eşitliği $0 \le r < n$ ve $0 \le q < n$ olmak üzere $g^r = h \cdot u^q$ olarak yeniden yazılabilir. Böylece g^r, 2. adımdaki listede ve $h \cdot u^q$, 3. adımdaki listededir. Bu ise iki liste arasında bir eşleme olduğunu gösterir.

2.4.4. Örnek. $g = 9704$, $h = 13896$ ve $p = 17389$ için Shanks'in bebek adımı-dev adımı yöntemini kullanarak $\mathbb{F}_p^*$ da $g^x = h$ ayrık logaritma problemini çözelim. 9704 sayısının $\mathbb{F}^*_{17389}$ daki mertebesi 1242'dir. $n = 1 + \lfloor \sqrt{1242} \rfloor = 36$ ve $u = g^{-n} = 9704^{-36} = 2494$ alınır. $\mathbb{F}^*_{17389}$ da, $k = 7$ için

$$g^k = 9704^7 = 14567$$

ve $l = 32$ için

$$h \cdot u^l = 13896 \cdot 2494^{32} = 14567$$

yani $9704^7 = 14567 = 13896 \cdot 2494^{32}$ ortak değeri bulunur. $\mathbb{F}^*_{17389}$ da, $2494 = 9704^{-36}$ olduğu kullanılırsa

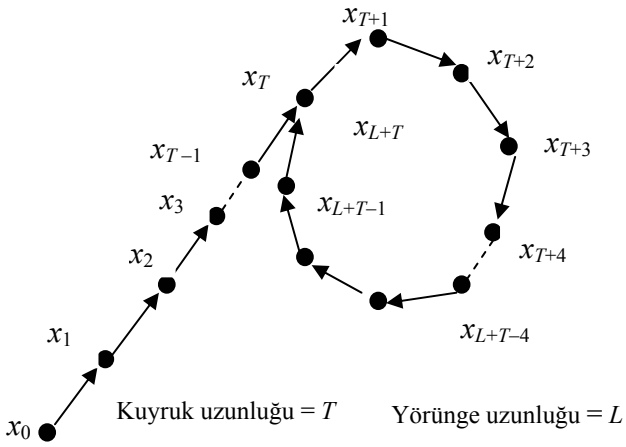
$$13896 = 9704^{7} \cdot 2494^{-32} = 9704^{7} \cdot (9704^{36})^{32} = 9704^{1159}$$

eşitliği elde edilir. Böylece $\mathbb{F}^{*}_{17389}$ daki $9704^{x} = 13896$ probleminin çözümü $x = 1159$ olarak bulunur (Hoffstein ve ark. 2008).

2.5. Pollard'ın ρ Algoritması

Shanks'in bebek adımı-dev adımı algoritması çok fazla depolama alanı gerektirir. Bu kısımda ele alınacak olan Pollard'ın ρ algoritması da bir çarpışma algoritmasıdır. Bu algoritmada daha az veri depolanır ve adım sayısı Shanks'in bebek adımı-dev adımı algoritması ile yaklaşık olarak aynıdır. Pollard'ın ρ algoritması, eliptik eğrilerle ayrık logaritma problemini çözmek için bilinen en iyi yöntemdir. Bu algoritma aşağıdaki çarpışma teoremine dayanmaktadır.

2.5.1. Teorem. S, N elemanlı bir sonlu küme ve $f: S \to S$ bir fonksiyon olsun. Başlangıç terimi $x_0 \in S$ olmak üzere $x_0, x_1, \ldots$ noktalarının bir dizisi $x_i = f(x_{i-1}) = f \circ f \circ \cdots \circ f(x_0)$ biçiminde tanımlansın. $i \geq 0$ olmak üzere (x_i) dizisinde x_{T-1} sadece bir kere görünecek şekilde en büyük tamsayı T ve $x_{T+L} = x_T$ olacak şekilde en küçük tamsayı L olsun.



Şekil 2.1. Pollard'ın ρ algoritmasındaki x_0 ın yörüngesi (Silverman 2009)

a) $x_{2i} = x_i$ olacak şekilde bir $1 \le i < T + L$ indeksi vardır.

b) $f : S \to S$ olmak üzere f fonksiyonunun iterasyonları (ardışık olarak uygulanması) S kümesinin elemanlarını karıştıracak kadar "yeterince rastgele" ise $T + L$ nin beklenen değeri $\sqrt{\pi N / 2}$ dir (Silverman 2009).

İspat. a) Şekle dikkat edilirse $j > i$ için $x_j = x_i \Leftrightarrow i \ge T$ ve $j \equiv i \pmod{L}$ olduğu açıktır. Böylece, $x_{2i} = x_i \Leftrightarrow i \ge T$ ve $L \mid i$ dir. Bu özellikteki ilk i, T ve $T + L - 1$ arasındadır.

b) S kümesinden rastgele seçilen k tane $x_0, x_1, x_2, \ldots, x_{k-1}$ noktalarının birbirinden farklı olma olasılığı

$$\text{Prob}(x_0, \ldots, x_{k-1} \text{ birbirinden farklı}) = \prod_{i=1}^{k-1} \text{Prob (her } 0 \le j < i \text{ için } x_i \ne x_j \mid x_0, \ldots, x_{i-1} \text{ farklı})$$

$$= \prod_{i=1}^{k-1} \left(\frac{N-i}{N} \right) = \prod_{i=1}^{k-1} \left(1 - \frac{i}{N} \right)$$

dir. t nin küçük değerleri için $1 - t \approx e^{-t}$ olduğu gerçeği kullanılırsa son çarpımın yaklaşık değeri

$$\text{Prob}(x_0, x_1, \ldots, x_{k-1} \text{birbirinden farklı}) \approx \prod_{i=1}^{k-1} \left(\frac{N-i}{N} \right) \approx e^{-k^2/2N}$$

olarak bulunur.

Şimdi $x_0, x_1, \ldots, x_{k-1}$ in birbirinden farklı olduğunu varsayalım. x_k noktasının önceki değerlerden biriyle eşleşme olasılığı,

$$\text{Prob}(x_k \text{ bir eşlemedir} \mid x_0, x_1, \ldots, x_{k-1} \text{birbirinden farklı}) = \frac{k}{N}$$

dir. Bu iki olasılık hesabı göz önüne alınırsa

$$\text{Prob}(x_k \text{ birinci eşlemedir}) = \text{Prob}(x_k \text{ bir eşleme ve } x_0, x_1, \ldots, x_{k-1} \text{ birbirinden farklı})$$

$$= \text{Prob}(x_k \text{ bir eşleme ve } x_0, x_1, \ldots, x_{k-1} \text{ birbirinden farklı})$$

$$\times \text{Prob}(x_0, x_1, \ldots, x_{k-1} \text{ birbirinden farklı})$$

$$\approx \frac{k}{N}\, e^{-k^2/2N}$$

olduğu elde edilir.

Böylece ilk eşlemeden önce beklenen adım sayısı

$$\sum_{k\geq 1} k \cdot \text{Prob}(x_k \text{ ilk eşlemedir}) \approx \sum_{k\geq 1} \frac{k^2}{N} \cdot e^{-k^2/2N}$$

dir. Diğer yandan $\Phi(t) = t^2\ e^{-t^2/2N}$ olarak alınır, $\sum_{k=1}^{\infty} \phi(k/n) \approx n\int_0^{\infty} \phi(t)$ olduğu kullanılırsa

$$\sum_{k\geq 1} k \cdot \text{Prob}(x_k \text{ ilk eşlemedir}) \approx \sqrt{N}\int_0^{\infty} t^2 e^{-t^2/2} = \sqrt{\pi N/2}$$

olarak bulunur.

2.5.2. Pollard'ın ρ Algoritması. G bir grup ve $x, y \in G$ olsun. $x^m = y$ eşitliğini gerçekleyen bir m tamsayısını bulmak için

$$x^i y^j = x^k y^l$$

olacak biçimde i, j, k, l tamsayılarını bulmak için Teorem 2.5.1' i kullanalım. $x^{i-k} = y^{j-l}$ olduğundan $j - l$ sayısının x in mertebesi olan n sayısı ile aralarında asal olduğu varsayılırsa y, x in bir kuvveti olarak çözülebilir.

$f : G \to G$ fonksiyonu, G grubundaki elemanları yeterince karıştırabilecek ve yörüngeleri kolay izlenebilecek biçimde tanımlanmalıdır. Pollard, G grubunu yaklaşık olarak eşit büyüklükte üç ayrık kümeden oluşacak biçimde parçalamıştır. Buna göre, $G = A \cup B \cup C$ olmak üzere

$$f(z) = \begin{cases} xz, & z \in A \\ z^2, & z \in B \\ yz, & z \in C \end{cases}$$

fonksiyonu kullanılarak işlemler yapılır.

f fonksiyonu tekrar tekrar başlangıç noktası $z_0 = 1$ noktasına uygulandığında f nin i iterasyonu sonrasında bir z_i noktası elde edilir. Böylece belli α_i, β_i tamsayıları için

$$z_i = f \circ f \circ \cdots \circ f(1) = x^{\alpha_i} y^{\beta_i}$$

olarak yazılabilir. Burada α_i, β_i değerleri $\alpha_0 = \beta_0 = 0$ olmak üzere z_1, z_2, … değerlerini hesaplayarak ve

$$\alpha_{i+1} = \begin{cases} \alpha_{i+1}, & z_i \in A \\ 2\alpha_i, & z_i \in B \\ \alpha_i, & z_i \in C \end{cases}, \qquad \beta_{i+1} = \begin{cases} \beta_i, & z_i \in A \\ 2\beta_i, & z_i \in B \\ \beta_{i+1}, & z_i \in C \end{cases}$$

iterasyon formüllerini kullanarak hesaplanır. Dikkat edilirse $x^n = 1$ olduğundan α_i ve β_i değerleri n modülüne göredir.

Benzer şekilde

$$w_0 = 1 \quad \text{ve} \quad w_{i+1} = f(f(w_i))$$

noktalarının dizisi hesaplanabilir. Böylece,

$$w_i = z_{2i} = x^{\gamma_i} y^{\delta_i}$$

olduğu elde edilir. Burada γ_i ve δ_i değerleri, ilk olarak $w_i = z_{2i}$ eşitliği ve daha sonra $f(w_i) = z_{2i+1}$ fonksiyonunu kullanarak α_i ve β_i için verilen bağıntılar yardımıyla γ_{i-1} ve δ_{i-1} değerlerinden hesaplanabilir.

Bu işlemler, (z_1, w_1), (z_2, w_2), … sıralı ikililerinden x ve y koordinatları birbirine eşit olan bir çift bulana kadar devam ettirilir. Dikkat edilirse her bir (z_i, w_i) sıralı ikilisi yalnızca önceki (z_{i-1}, w_{i-1}) sıralı ikilisi ile elde edilebileceğinden depolanan veri azdır. A, B, C kümelerinin G grubunun elemanlarının karıştırılmasında yeterince iyi olduğu varsayılırsa bir önceki teorem gereği, $O(\sqrt{N})$ adımda

$$z_i = w_i = z_{2i}$$

şeklinde bir eşleme bulunur. $z_i = w_i$ olduğundan

$$x^{\alpha_i - \gamma_i} = y^{\delta_i - \beta_i}$$

eşitliği elde edilir. n bir asal sayı olmak üzere $(\delta_i - \beta_i, n) = 1$ ise

$$m \equiv (\alpha_i - \gamma_i)\,(\delta_i - \beta_i)^{-1} \pmod{n}$$

olarak bulunur. Bu ise $x^m = y$ eşitliğini çözer.

Genel olarak, $d = (\delta_i - \beta_i, n) > 1$ ise y^d, x in bir kuvveti olarak ifade edilebilir, örneğin $y^d = x^e$ olarak alınabilir. Bu durumda $0 \le u < d$ olmak üzere y, $x^{(e+nu)/d}$ elemanlarından birine eşittir. Dolayısıyla eğer d büyük bir sayı değilse, bu ayrık logaritma problemini çözer, eğer d büyükse Pollard'ın algoritması x ve y arasında bir ilişki bulana kadar çalışır.

2.5.3. Uyarı. Shanks'in ve Pollard'ın algoritmaları herhangi bir gruba uygulanabilir ve bu algoritmalar, n mertebeli devirli bir G grubundaki ayrık logaritma probleminin $O(\sqrt{N})$ adımda çözülebileceğini gösterir. G grubunun işlemlerini gerçekleştiren bir kara kutu olduğu düşünülürse gruptaki herhangi x_1, x_2 elemanları bu kutuya atıldığında x_1x_2 değerini hesaplar, ancak kara kutuda yapılan bu hesaplamanın nasıl olduğu bilinmez. Shoup (1997), kara kutu algoritması yardımıyla, G grubunda ayrık logaritma problemini çözen herhangi bir algoritmanın en az $O(\sqrt{N})$ adım sürdüğünü göstermiştir. Dolayısıyla eliptik eğri üzerindeki grup yapısı kullanılarak ayrık logaritma problemini çözmek için bilinen en iyi algoritmalar bile kara kutu algoritmasından daha iyi değildir.

2.6. Pohlig-Hellman Algoritması

G bir grup ve $g \in G$ mertebesi N olan bir eleman olmak üzere G grubundaki $g^x = h$ eşitliğinin çözümü N modülüne göre belirlenebileceğinden N sayısının asal çarpanlaması oldukça önemlidir. Aşağıda verilecek olan Pohlig-Hellman algoritması N sayısının asal çarpanlarını kullanır. Bu algoritmayı vermeden önce aşağıdaki uyarıya ihtiyaç vardır.

2.6.1. Uyarı. G bir grup g, $h \in G$ ve q bir asal sayı olmak üzere $g \in G$ elemanının mertebesi q^e olsun. G grubunda $g^x = h$ ALPni çözen bir algoritma olduğunu varsayalım.

Bu durumda ALP, $O(S_q^e)$ adımda çözülebilir. Eğer e, q dan çok daha küçük bir sayı ise ALP, $S_q^e = e\sqrt{q}$ adımda çözülür, eğer Shanks'in bebek adımı-dev adımı algoritması kullanılırsa $S_q^e = q^{e/2}$ dir.

2.6.2. Teorem. G bir grup olmak üzere $g \in G$ elemanının mertebesi N olsun ve N sayısı asal sayıların kuvvetlerinin çarpımı biçiminde

$$N = \prod_{i=1}^{t} q_i^{e_i}$$

olarak yazılsın. Bu durumda $g^x = h$ ayrık logaritma problemi

$$O(\sum_{i=1}^{t} S_{q_i^{e_i}} + \log N)$$

adımda aşağıdaki gibi çözülebilir.

1) Her $1 \le i \le t$ için

$$g_i = g^{N/q_i^{e_i}} \qquad \text{ve} \qquad h_i = h^{N/q_i^{e_i}}$$

olsun. g_i elemanının mertebesi $q_i^{e_i}$ asal kuvveti olduğundan

$$g_i^y = h_i$$

ayrık logaritma problemini çözmek için uyarıda geçen algoritmalardan biri kullanılır ve bu eşitliğin bir çözümü $y = y_i$ olarak alınır.

2) Çinlilerin kalan teoremi kullanılarak

$$x \equiv y_1 \pmod{q_1^{e_1}},\ x \equiv y_2 \pmod{q_2^{e_2}},\ \ldots,\ x \equiv y_t \pmod{q_t^{e_t}}$$

denkliklerinin çözümü bulunur (Hoffstein ve ark. 2008).

İspat. (1) adımı $O(\sum_{i=1}^{t} S_{q_i^{e_i}})$ zaman, (2) adımı ise Çinlilerin kalan teoremi kullanarak $O(\log N)$ zaman alır. Dolayısıyla algoritmanın çalışma süresinin $O(\sum_{i=1}^{t} S_{q_i^{e_i}} + \log N)$ olduğu açıktır.

Şimdi (1) ve (2) adımlarının $g^x = h$ ayrık logaritma probleminin bir çözümünü verdiğini görelim. x, (2) adımındaki denklik sisteminin bir çözümü olsun. O halde her i ve belli z_i ler için

$$x = y_i + q_i^{e_i} z_i$$

olarak yazılabilir. Bu eşitlikten

$$(g^x)^{N/q_i^{e_i}} = (g^{y_i + q_i^{e_i} z_i})^{N/q_i^{e_i}} = (g^{N/q_i^{e_i}})^{y_i} g^{Nz_i}$$

eşitliği elde edilir. Diğer yandan $g^N = 1$ ve $g_i = g^{N/q_i^{e_i}}$, $g_i^y = h_i$ ve son olarak $h_i = h^{N/q_i^{e_i}}$ olduğu hatırlanırsa

$$(g^x)^{N/q_i^{e_i}} = g_i^{y_i} = h_i = h^{N/q_i^{e_i}}$$

olduğu elde edilir.

g^N elemanı etkisiz eleman olduğundan g tabanında ayrık logaritma sadece N modülüne göre tanımlıdır. Böylece son eşitlikten g tabanında ayrık logaritma problemi

$$\frac{N}{q_i^{e_i}} x \equiv \frac{N}{q_i^{e_i}} \log_g(h) \pmod{N} \qquad (2.\ 1)$$

olarak yeniden yazılabilir. Dikkat edilirse $\frac{N}{q_1^{e_1}}$, … , $\frac{N}{q_t^{e_t}}$ sayılarının ortak çarpanı yoktur, yani bu sayılar aralarında asaldır. Dolayısıyla

$$\frac{N}{q_1^{e_1}} \cdot c_1 + \cdots + \frac{N}{q_t^{e_t}} \cdot c_t = 1 \qquad (2.\ 2)$$

olacak şekilde $c_1, c_2, \ldots, c_t$ tamsayıları vardır. (2. 1) denkliğinin her iki tarafı c_i lerle çarpılır ve $i = 1, 2, \ldots, t$ üzerinden toplam alınırsa

$$\sum_{i=1}^{t} \frac{N}{q_i^{e_i}} c_i x \equiv \sum_{i=1}^{t} \frac{N}{q_i^{e_i}} c_i \log_g(h) \pmod{N}$$

denkliği elde edilir. Böylece (2. 2) eşitliği dikkate alınırsa

$$x \equiv \log_g(h) \pmod{N}$$

denkliği elde edilir.

2.6.3. Uyarı. Pohlig-Hellman algoritmasına göre, G grubunun mertebesi küçük asal sayıların kuvvetinin çarpımı biçiminde yazılabilirse G grubunda ayrık logaritma problemi güvenli değildir. Diğer bir ifade ile, g elemanının mertebesi küçük asal sayıların kuvvetinin bir çarpımı ise $g^x = h$ eşitliğini çözmek kolaydır. Bu durum özellikle $p - 1$ sayısının çarpanlarının küçük asal sayıların kuvvetleri biçiminde yazılabilmesi halinde $\mathbb{F}_p$ deki ayrık logaritma problemi için geçerlidir. Bu halde $p - 1$ çift olduğundan q bir asal sayı olmak üzere $p = 2q + 1$ ve q mertebeli bir g elemanı alınabilir. Bu algoritmanın çalışma süresi Önerme 2.4.3 gereği, $O(\sqrt{q}) = O(\sqrt{p})$ dir. Bununla birlikte daha sonraki kısımda görüleceği gibi indeks hesabı yönteminin çalışma süresi alt üsteldir. Dolayısıyla $p = 2q + 1$ olarak alınsa bile q asal sayısı oldukça büyük seçilmelidir.

Aşağıdaki önermede, mertebesi asal sayıların kuvvetleri olan elemanlar için ayrık logaritma probleminin mertebesi asal elemanlar için ayrık logaritma problemine indirgendiği görülecektir. Bu önerme g elemanının mertebesi q^e ise $g^{q^{e-1}}$ elemanının mertebesinin q olduğunu kullanır. Teorem 2.6.2 deki notasyonlarla, aşağıda da görüleceği gibi, q^e mertebeli elemanlar için çalışma süresi $S_q^{\,e}$ den $O(eS_q)$ ya indirgenebilir. Önerme, G bir grup g, $h \in G$ ve q bir asal sayı, $g \in G$ elemanının mertebesi q olmak üzere G grubunda $g^x = h$ ALPni S_q adımda çözen bir algoritma olduğunu varsaymaktadır. Dolayısıyla Shanks'in bebek adımı-dev adımı algoritması kullanılırsa ALP, $S_q = O(\sqrt{q})$ adımda çözülebilir, ancak bu önerme $g \in G$ elemanının mertebesi q^e ise ALP nin $O(e\sqrt{q})$ adımda çözülebildiğini ifade etmektedir.

2.6.4. Önerme. G bir grup, q bir asal sayı ve $a \in G$ elemanın mertebesi q olmak üzere G grubundaki $a^x = b$ ayrık logaritma problemini S_q adımda çözen bir algoritma olduğunu varsayalım. $e \geq 1$ olmak üzere $g \in G$ elemanının mertebesi q^e ise $g^x = h$ ayrık logaritma problemi $O(eS_q)$ adımda çözülebilir (Hoffstein ve ark. 2008).

İspat. $g^x = h$ eşitliğindeki x kuvveti, $0 \leq x_i < q$ olmak üzere

$$x = x_0 + x_1 q + x_2 q^2 + \cdots + x_{e-1} q^{e-1} \qquad (2.\ 3)$$

olacak şekilde yazılabilir ve $x_0, x_1, x_2, \ldots$ değerleri belirlenebilir. $g^x = h$ eşitliğinin her iki tarafının q^{e-1} inci kuvveti alınır ve (2. 3) eşitliği kullanılırsa

$$h^{q^{e-1}} = (g^x)^{q^{e-1}} = (g^{x_0 + x_1 q + \ldots + x_{e-1} q^{e-1}})$$

ve bu eşitlik yeniden düzenlenir ve $g^{q^e} = 1$ olduğu kullanılırsa

$$h^{q^{e-1}} = g^{x_0 q^{e-1}} (g^{q^e})^{x_1 + \ldots + x_{e-1} q^{e-2}} = (g^{q^{e-1}})^{x_0}$$

eşitliği elde edilir. $g^{q^{e-1}}$, G grubunda q mertebeli bir eleman olduğundan

$$(g^{q^{e-1}})^{x_0} = h^{q^{e-1}}$$

eşitliği tabanı q mertebeli bir eleman olan bir ayrık logaritma problemidir. Varsayım gereği, bu ayrık logaritma problemi S_q adımda çözülebilir. Bu durumda G grubunda

$$g^{x_0 q^{e-1}} = h^{q^{e-1}}$$

olacak biçimde x_0 üssü bilindiğinden benzer bir hesaplama yapılabilir. Buna göre $g^x = h$ eşitliğinin her iki tarafının q^{e-2} inci kuvveti alınır ve gerekli düzenlemeler yapılırsa

$$h^{q^{e-2}} = (g^x)^{q^{e-2}} = (g^{x_0 + x_1 q + \ldots + x_{e-1} q^{e-1}})^{q^{e-2}} = g^{x_0 q^{e-2}} g^{x_1 q^{e-1}}$$

eşitliği elde edilir. Dikkat edilirse x_0 değeri belirlenmiştir ve $g^{q^{e-1}}$ elemanının G grubundaki mertebesi q dur. x_1 değerini bulmak için

$$(g^{q^{e-1}})^{x_1} = (h g^{-x_0})^{q^{e-2}}$$

ayrık logaritma problemi çözülmelidir. Verilen algoritma tekrar uygulanırsa bu problem S_q adımda çözülebilir. Böylece $O(2S_q)$ adımda, G grubunda

$$(g^{x_0 + x_1 q})^{q^{e-2}} = h^{q^{e-2}}$$

eşitliğini gerçekleyen x_0 ve x_1 değerleri belirlenmiş olur.

Benzer şekilde

$$(g^{q^{e-1}})^{x_2} = (hg^{-x_0 - x_1 q})^{q^{e-3}}$$

ayrık logaritma problemi çözülerek x_2 değeri bulunur. Bu şekilde devam edilerek x_0, … , x_{i-1} değerleri belirlendikten sonra G grubunda

$$(g^{q^{e-1}})^{x_i} = (hg^{-x_0 - x_1 q - \ldots - x_{i-1} q^{i-1}})^{q^{e-i-1}}$$

ayrık logaritma problemi çözülerek x_i değeri bulunur.

Bunların her biri tabanı q mertebeli olan bir ayrık logaritma problemidir ve varsayım gereği her bir problem S_q adımda çözülebilir. Dolayısıyla $O(eS_q)$ adımda $g^x = h$ eşitliğini gerçekleyen bir $x = x_0 + x_1 q + x_2 q^2 + \cdots + x_{e-1} q^{e-1}$ kuvveti bulunur. Böylece ayrık logaritma problemi çözümü elde edilmiş olur.

2.6.5. Örnek. $\mathbb{F}^*_{11251}$ grubunda $5448^x = 6909$ ayrık logaritma problemini çözelim. Buna göre 11250 sayısı 5^4 ile bölündüğünden $\mathbb{F}_{11251}$ grubunda 5448 elemanının mertebesinin 5^4 dür. Bu ayrık logaritma probleminin çözümü için ilk olarak

$$(5448^{5^3})^{x_0} = 6909^{5^3}$$

eşitliğini alalım. Dikkat edilirse bu eşitlik $\mathbb{F}^*_{11251}$ grubunda $11089^{x_0} = 11089$ eşitliğine indirgenir ve böylece eşitliğin çözümü $x_0 = 1$ dir. Böylece x bilinmeyen kuvvetinin ilk değeri $x_0 = 1$ olarak bulunur.

İkinci olarak

$$(5448^{5^3})^{x_1} = (6909 \cdot 5448^{-x_0})^{5^2} = (6909 \cdot 5448^{-1})^{5^2}$$

eşitliğini alalım. Bu eşitlik $\mathbb{F}^*_{11251}$ grubunda $11089^{x_1} = 3742$ eşitliğine indirgenir. Eğer q büyük bir sayı olsaydı bu ayrık logaritma problemini çözmek için Shanks'in bebek adımı-dev adımı gibi hızlı bir algoritma kullanılabilir, ancak bu durumda x_1 sayısının 1 ve 4 arasındaki değerlerini denemek yeterlidir. Böylece $x_1 = 2$ çözümü bulunur. Bulunan x_1 değeriyle birlikte $x = 11 = 1 + 2 \cdot 5$ olduğu elde edilir.

Şimdi x_2 değerini bulmak için

$$(5448^{5^3})^{x_2} = (6909 \cdot 5448^{-x_0-x_1\cdot 5})^5 = (6909 \cdot 5448^{-11})^5$$

eşitliği $\mathbb{F}^*_{11251}$ grubunda $11089^{x_2} = 1$ eşitliğine indirgenir ve dolayısıyla $x_2 = 0$ olarak bulunur. Böylece x değeri değişmez, yani $x = 1 + 2 \cdot 5 + 0 \cdot 5^2 = 11$ dir.

Son olarak

$$(5448^{5^3})^{x_3} = (6909 \cdot 5448^{-x_0-x_1\cdot 5-x_2\cdot 5^2}) = 6909 \cdot 5448^{-11}$$

eşitliği $\mathbb{F}^*_{11251}$ grubunda $11089^{x_3} = 6320$ eşitliğine indirgenir ve böylece $x_3 = 4$ olduğu elde edilir. O halde

$$x = 511 = 1 + 2 \cdot 5 + 4 \cdot 5^3$$

dir. Dikkat edilirse

$$5448^{511} \equiv 6909 \pmod{11251}$$

dur (Hoffstein ve ark. 2008).

2.7. $\mathbb{F}_p$ deki Ayrık Logaritma Problemini Hesaplamak İçin İndeks Hesabı Yöntemi

İndeks hesabı yöntemi, açık anahtar kriptolojisinin bulunuşundan birkaç yıl önce ilk olarak 1968'de Western ve Miller'in (1968) çalışmasında ortaya çıkmıştır. Bu yöntem, Diffie-Hellman'nın (1976) makalesinin yayınlanmasından sonra 1970'li yıllarda birkaç kriptosistemci tarafından yeniden keşfedilmiştir. İndeks hesabı, $\mathbb{F}_p$ sonlu cismindeki ayrık logaritma probleminin çözümü için kullanılan bir yöntemdir. Aşağıda bu yöntemde kullanılacak olan bir tanım verilmiştir.

2.7.1. Tanım. n bir tamsayı olsun. Eğer n, bir B sayısından küçük ya da eşit olan asal sayıların çarpımı biçiminde yazılabiliyorsa n tamsayısına *B-düzgün* (*B-smooth*) *sayı* denir.

Örneğin, 18 sayısı 3-düzgün sayıdır. Gerçekten de, $18 = 2 \cdot 3^2$ olduğundan 18 sayısı 2 ve 3 asal sayılarının kuvvetleri biçiminde yazılır.

2.7.2. İndeks Hesabı Yöntemi. p asal sayı ve g, h tamsayılar olmak üzere

$$g^x \equiv h \pmod{p}$$

ayrık logaritma problemini çözmek için g yi p modülüne göre bir ilkel kök olarak alalım. Bu yöntemde $g^x \equiv h \pmod{p}$ ayrık logaritma problemini doğrudan çözmek yerine bir B düzgün sayısı seçilir ve $l \leq B$ özelliğindeki her l asal sayısı için

$$g^x \equiv l \pmod{p}$$

ayrık logaritma problemi çözülür. Diğer bir deyişle $l \leq B$ özelliğindeki her l asal sayısı için $\log_g(l)$ ayrık logaritması hesaplanır.

Daha sonra $k = 1, 2, \ldots$ için $h \cdot g^{-k} \pmod{p}$ değerleri hesaplanır, bu işleme $h \cdot g^{-k} \pmod{p}$ değerleri B-düzgün sayı olacak şekilde bir k değeri bulana kadar devam edilir. Bu özellikteki k sayısı belli e_l ler için

$$h \cdot g^{-k} \equiv \prod_{l \leq B} l^{e_l} \pmod{p} \tag{2. 4}$$

denkliği elde edilir. Bu denklik

$$\log_g(h) \equiv k + \sum_{l \leq B} e_l \log_g(l) \pmod{p-1} \tag{2. 5}$$

olarak yeniden yazılabilir, burada ayrık logaritmalar sadece $p - 1$ modülüne göre tanımlıdır. $l \leq B$ özelliğindeki her l asal sayısı için $\log_g(l)$ değerinin hesaplandığı varsayıldığından (2. 5) denkliği $\log_g(h)$ değerini verir.

Şimdi küçük l asal sayıları için $\log_g(l)$ değerinin nasıl bulunduğunu açıklayalım. $0 < g_i < p$ olmak üzere i nin rastgele seçimi için

$$g_i \equiv g^i \pmod{p}$$

denkliği hesaplanır.

g_i, B-düzgün değilse bu g_i değeri alınmaz ve elenir, eğer g_i değeri B-düzgün ise g_i

$$g_i = \prod_{l \le B} l^{u_l(i)}$$

olarak çarpanlarına ayrılabilir. Böylece

$$i \equiv \log_g(g_i) \equiv \sum_{l \le B} u_l(i) \cdot \log_g(l) \pmod{p-1} \qquad (2.\ 6)$$

denkliği elde edilir. Dikkat edilirse (2. 6) denkliğinde bilinmeyen değerler $\log_g(l)$ ayrık logaritma değerleridir. B sayısından küçük veya B sayısına eşit olan asal sayıların eleman sayısı $\pi(B)$ ile gösterilirse, (2. 6) denkliğine benzer $\pi(B)$ sayısından daha fazla denklik bulunabilirse $\log_g(l)$ değerlerini bulmak için lineer cebir kullanılabilir.

2.7.4. Örnek. p = 18443 olmak üzere indeks hesabı yöntemini kullanarak

$$37^x \equiv 211 \pmod{18443}$$

ayrık logaritma problemini çözelim. g = 37 sayısı p = 18443 modülüne göre bir ilkel köktür. Eğer B = 5 olarak alınırsa asal sayıların oluşturduğu çarpım tabanı kümesi {2, 3, 5} tir. 18443 modülüne göre g = 37 sayısının rastgele kuvvetlerini alınır ve B-düzgün sayıları seçilirse

$$g^{12708} \equiv 2^3 \cdot 3^4 \cdot 5 \pmod{18443}, \qquad g^{15400} \equiv 2^3 \cdot 3^3 \cdot 5 \pmod{18443}$$

$$g^{11311} \equiv 2^3 \cdot 5^2 \pmod{18443}, \qquad g^{2731} \equiv 2^3 \cdot 3 \cdot 5^4 \pmod{18443}$$

denklikleri elde edilir. Bu denklikler yardımıyla g tabanına göre 2, 3 ve 5'in ayrık logaritmaları için eşitlikler elde edilir. Örneğin, birinci denklikten

$$12708 = 3 \cdot \log_g(2) + 4 \cdot \log_g(3) + \log_g(5)$$

eşitliği elde edilir. Eğer

$$x_2 = \log_g(2),\ x_3 = \log_g(3),\ x_5 = \log_g(5)$$

denirse yukarıdaki dört denklikten

$$12708 = 3x_2 + 4x_3 + x_5$$

$$15400 = 3x_2 + 3x_3 + x_5$$

$$11311 = 3x_2 + 2x_5$$

$$2731 = 3x_2 + x_3 + 4x_5 \qquad (2.\ 7)$$

lineer denklemleri elde edilir. Dikkat edilirse ayrık logaritmalar sadece $p - 1$ modülüne göre tanımlı olduğundan Böylece (2. 7) deki formüller $p - 1 = 18442 = 2 \cdot 9221$ modülüne göredir. 9221 sayısı bir asal sayı olduğundan (2. 7) deki lineer denklem sisteminin 2 ve 9221 modülüne göre çözülmesi gerekir. Bu lineer denklem sistemi Gauss yöntemiyle kolaylıkla çözülebilir. x_2, x_3, x_5 bilinmeyenlerinin 2 modülüne ve 9221 modülüne göre çözümleri

$$(x_2, x_3, x_5) \equiv (1, 0, 1) \pmod 2$$
$$(x_2, x_3, x_5) \equiv (5733, 6529, 6277) \pmod{9221}$$

olduğundan

$$(x_2, x_3, x_5) \equiv (5733, 15750, 6277) \pmod{18442}$$

olduğu elde edilir. Daha sonra

$$37^{5733} \equiv 2 \pmod{18443},\ 37^{15750} \equiv 3 \pmod{18443},\ 37^{6277} \equiv 5 \pmod{18443}$$

değerleri hesaplanarak çözümler kontrol edilir.

$37^x \equiv 211 \pmod{18443}$ ayrık logaritma probleminin çözümü bulunmak istendiğinden k sayısının rastgele değerleri için B-düzgün $211 \cdot 37^{-k} \pmod{18443}$ değerini bulana kadar hesaplama yapılırsa

$$211 \cdot 37^{-9549} \equiv 2^5 \cdot 3^2 \cdot 5^2 \pmod{18443}$$

olduğu elde edilir. Yukarıda elde edilen 2, 3 ve 5'in ayrık logaritma değerleri kullanılarak

$$\log_g(211) = 9549 + 5\log_g(2) + 2\log_g(3) + 2\log_g(5)$$
$$= 9549 + 5 \cdot 5733 + 2 \cdot 15750 + 2 \cdot 6277 \equiv 8500 \pmod{18442}$$

olduğu elde edilir. Son olarak $37^{8500} \equiv 211 \pmod{18443}$ olduğundan $\log_g(211) = 8500$ dür (Hoffstein ve ark. 2008).

2.7.5. Uyarı. İndeks hesabının çalışma süresi kabaca hesaplanabilir. B sayısından küçük asal sayılardan oluşan bir çarpım tabanı kullanılarak yaklaşık $\pi(B)$ tane B-düzgün $g^i \pmod p$ sayıları bulunmalıdır. Bunun için elek-tipi (sieve-type) gibi çeşitli yöntemler

kullanılabilir. Her durumda indeks hesabı, $\mathbb{F}_p^*$ daki ayrık logaritma problemini çözmek için bir altüstel algoritmadır. Ancak daha sonra da görülebileceği gibi eliptik eğri gruplarındaki genel ayrık logaritma problemini çözmek için bilinen en iyi algoritmalar üstel algoritmalardır.

3. MATERYAL VE YÖNTEM

Çalışmada eliptik eğri ayrık logaritma problemi daha kolay bir ayrık logaritma problemine dönüştürülmek istendiğinden eliptik eğriler kavramı oldukça önemlidir. Dolayısıyla bu bölümde eliptik eğriler ve eliptik eğriler ile ilgili temel kavramlar ele alınacak, eliptik eğrilerin ayrık logaritma probleminde nasıl kullanıldığını gösteren bazı temel teoremler incelenecektir. Daha sonra eliptik eğri ayrık logaritma problemi üzerinde durulacak ve bu problemin çözümü için bazı algoritmalar verilecektir.

Eliptik eğriler, Fermat'nın son teoreminin ispatında A. Wiles tarafından kullanıldıktan sonra oldukça popüler olmuşlardır. Böylece eliptik eğriler teorisi, cebirin de önemli bir alanı haline gelmiştir. Son yıllarda eliptik eğriler kriptosistemciler tarafından şifreleme yöntemlerinde kullanılmaya başlanmıştır. Böylece bu şekilde oluşturulan şifrelerin kırılmasında son teknoloji bilgisayarların kullanılması halinde bile uzun zaman alması hedeflenmiştir.

3.1. Eliptik Eğriler

Bu kısımda eliptik eğriler tanıtılacak ve bu eğrilerin temel özellikleri ele alınacaktır. Eliptik eğriler ile ilgili detaylı bilgi için (Silverman 2009) ve (Silverman ve Tate, 2015) kitapları incelenebilir.

3.1.1. Tanım. $\mathbb{F}$ karakteristiği 2 ve 3 ten farklı bir cisim, $a, b \in \mathbb{F}$ ve $4a^3 + 27b^2 \neq 0$ olmak üzere

$$E : y^2 = x^3 + ax + b$$

şeklindeki denklemin tüm çözümlerinin oluşturduğu (x, y) sıralı ikililerinin kümesine $\boldsymbol{O}$ noktası ile birlikte bir *eliptik eğri* denir. Bu denkleme E eliptik eğrisinin *Weierstrass formu* denir.

$\mathbb{F}$ bir cisim ve $a_1, a_2, a_3, a_4, a_6 \in \mathbb{F}$ olmak üzere

$$E: y^2 + a_1 x y + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6$$

biçimindeki bir denkleme E eliptik eğrisinin *uzun Weierstrass formu* denir.

Eğer E, $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri ise bu eğri üzerindeki noktaların kümesi $E(\mathbb{F})$ ile gösterilir, yani

$$E(\mathbb{F}) = \{(x, y) : x, y \in \mathbb{F} \times \mathbb{F} |\ y^2 = x^3 + ax + b\} \cup \{\boldsymbol{O}\}$$

dir. $E(\mathbb{F})$, E eliptik eğrisinin üzerinde tanımlanan özel bir toplama işlemine göre bir gruptur ve O noktası bu grubun etkisiz elemanıdır. $\boldsymbol{O}$ noktası sonsuzdaki nokta olarak adlandırılır ve bu noktanın x-eksenine dik olan tüm doğruların üzerinde olduğu varsayılır.

Bir E eliptik eğrisi üzerindeki toplama işlemi ise şu şekilde tanımlanır: Bir E eliptik eğrisi üzerinde iki farklı P ve Q noktaları toplamak için bu noktalardan geçen l doğrusunu alalım. l doğrusu E eliptik eğrisini üçüncü bir R noktasında keser, R noktasının x-eksenine göre simetriği olan nokta $P + Q = R'$ noktası olarak tanımlanır. Eğer P noktası kendisiyle toplanmak istenirse bu noktadan geçen teğet doğru dikkate alınır. Eğer bir $P = (x, y)$ noktasının x-eksenine göre simetriği olan $P' = (x, -y)$ noktası ile toplanmak istenirse P ve P' noktalarından geçen doğrusu x-eksenine diktir. Dolayısıyla

$$P + P' = \boldsymbol{O}$$

dur. Son olarak E eliptik eğrisi üzerinde bir P noktası ile $\boldsymbol{O}$ noktası toplanmak istenirse P noktası ile O noktasından geçen l doğrusu x-eksenine dik olduğundan $P + \boldsymbol{O} = P$ dir.

E, $y^2 = x^3 + ax + b$ Weierstrass denklemi ile verilen bir eliptik eğri olmak üzere $P = (x, y) \in E$ ise $-P = (x, -y)$ dir. Bundan başka, $P \in E$ ve $m \in \mathbb{Z}$ olmak üzere $m > 0$ ise

$$mP = \underbrace{P + \ldots + P}_{m \text{ tane}}$$

ve $m < 0$ ise $mP = (-m)(-P)$ olarak alınır ve $0P = \boldsymbol{O}$ olarak tanımlanır.

3.1.2. Teorem. E, $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri olmak üzere $E(\mathbb{F})$ abelyen bir gruptur (Silverman 2009).

Aşağıdaki teoremde E eliptik eğrisi üzerindeki toplama işlemi cebirsel olarak verilmiştir.

3.1.3. Teorem (Eliptik Eğriler için Toplama İşlemi Algoritması).

Input: $E : y^2 = x^3 + ax + b$ bir eliptik eğri ve E eğrisi üzerinde P_1 ve P_2 noktaları olsun.

1. Eğer $P_1 = \boldsymbol{O}$ ise $P_1 + P_2 = P_2$ dir.

2. $P_1 \neq \boldsymbol{O}$ ve $P_2 = \boldsymbol{O}$ ise $P_1 + P_2 = P_1$ dir.

3. $P_1, P_2 \neq \boldsymbol{O}$ ise $P_1 = (x_1, y_1)$ ve $P_2 = (x_2, y_2)$ olarak al.

4. Eğer $x_1 = x_2$ ve $y_1 = -y_2$ ise $P_1 + P_2 = \boldsymbol{O}$ dur.

5. $P_1 \neq P_2$ ve $x_1 \neq x_2$ ise $\lambda = \dfrac{y_2 - y_1}{x_2 - x_1}$, $P_1 = P_2$ ve $y_1 \neq 0$ ise $\lambda = \dfrac{3x_1^2 + a}{2y_1}$ olarak tanımla ve

$$x_3 = \lambda^2 - x_1 - x_2 \text{ ve } y_3 = \lambda (x_1 - x_3) - y_1$$

olarak al. Bu durumda $P_1 + P_2 = (x_3, y_3)$ tür (Hoffstein ve ark. 2008).

Bir E eliptik eğrisi üzerindeki büküm noktaları kümesinin kriptolojide oldukça önemli uygulamaları vardır.

3.1.4. Tanım. E, bir eliptik eğri ve $m \geq 1$ bir tamsayı olsun. E eliptik eğrisinin m mertebeli noktalarının kümesine E eliptik eğrisinin *m-büküm altgrubu* denir ve $E[m]$ ile gösterilir, yani

$$E[m] = \{P \in E \mid mP = \boldsymbol{O}\}$$

dir.

Aşağıdaki teorem $E[m]$ grubunun nasıl bir grup olduğunu belirtmektedir.

3.1.5. Teorem. E, $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri olmak üzere $m \geq 1$ bir tamsayı olsun. Eğer $\mathbb{K}$ cisminin karakteristiği m sayısını bölmüyor veya 0 ise

$$E[m] \cong \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$$

dir. Eğer K cisminin karakteristiği $p > 0$, $p \mid m$ ve $p \nmid m'$ olmak üzere $m = p^r m'$ ise

$$E[m] \cong \mathbb{Z}/m'\mathbb{Z} \times \mathbb{Z}/m'\mathbb{Z} \quad \text{veya } E[m] \cong \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m'\mathbb{Z}$$

dir (Washington 2003).

3.2. Bir Eliptik Eğrinin Bölüm Polinomları

Şimdi bir eliptik eğrinin bölüm polinomları kavramını ele alalım. E, bir $\mathbb{K}$ cismi üzerinde tanımlı ve Weierstrass denklemi

$$E: y^2 + a_1 x y + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6$$

olan bir eliptik eğri olsun. E eliptik eğrisinin F_n *bölüm polinomları*

$F_1 = 1$

$F_2 = 2y + a_1x + a_3$

$F_3 = 3x^4 + b_2x^3 + 3b_4x^2 + 3b_6x + b_8$

$F_4 = F_2(2x^6 + b_2x^5 + 5b_4x^4 + 10b_6x^3 + 10b_8x^2 + (b_2b_8 - b_4b_6)x + (b_4b_8 - b_6^2))$

başlangıç terimlerini kullanarak $m \geq 2$ için

$$F_{2m+1} = F_{m+2}F_m^3 - F_{m-1}F_{m+1}^3,$$

ve $m \geq 3$ için,

$$F_{2m}F_2 = F_{m-1}^2 F_m F_{m+2} - F_{m-2}F_m F_{m+1}^3$$

indirgeme bağıntıları ile elde edilir. Burada b_i terimleri E eliptik eğrisinin Tate değerleridir (Silverman 2009, Chapter III.1).

Bir eliptik eğrinin bölüm polinomları kavramı literatürde oldukça önemlidir. Bir eliptik eğrinin bölüm polinomları kavramı eliptik fonksiyonlar teorisi, eliptik bölünebilir diziler teorisi ile yakından ilgilidir. Bundan başka E, karakteristiği 2'den farklı bir $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri ve $P \in E(\mathbb{F})$ olmak üzere $n \geq 1$ tamsayısı için P noktasının n katı nP,

$$nP = \left(\frac{G_n(P)}{F_n(P)^2}, \frac{H_n(P)}{F_n(P)^3} \right) \tag{3.1}$$

olarak yazılabilir. Bundan başka G_n ve H_n bölüm polinomları

$G_0 = 1,\ G_1 = x$

$H_0 = 1,\ H_1 = y$ (3.2)

başlangıç terimlerini kullanarak $n \geq 2$ için

$G_n = xF_n^{\,2} - F_{n+1}F_{n-1},$

$H_n = (F_{n-1}^{\,2}F_{n+2} - F_{n-2}F_{n+1}^{\,2} - F_2F_n(a_1G_n + a_3F_n^{\,2}))(2F_2)^{-1}$ (3.3)

indirgeme bağıntıları ile elde edilirler.

Bir eliptik eğrinin F_n bölüm polinomları aşağıdaki zincir kuralını gerçekler.

3.2.1. Lemma. E, bir $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri ve $P \in E(\mathbb{F})$ olmak üzere her $m, n \geq 1$ tamsayıları için

$$F_{mn} = F_m^{n^2} \circ (F_n \circ m)$$

dir (Mazur ve Tate 1991).

3.3. Eliptik Eğri Üzerinde Bölenler

f bir rasyonel fonksiyon olmak üzere f fonksiyonunun sıfırları ve kutup yerleri katlılıkları da sayarak bir formal toplamla ifade edilebilir. Buna göre, f fonksiyonun birbirinden farklı sıfırları $\alpha_1, \ldots, \alpha_r$ ve bu sıfırların katlılıkları sırasıyla $d_1, \ldots, d_r$ ve f fonksiyonun birbirinden farklı kutupları $\beta_1, \ldots, \beta_s$ ve bu kutupların katlılıkları sırasıyla $e_1, \ldots, e_s$ olmak üzere f fonksiyonunun *böleni* (*divisor*) div(f) ile gösterilir ve

$$\text{div}(f) = d_1(\alpha_1) + \ldots + d_r(\alpha_r) - e_1(\beta_1) - \ldots - e_s(\beta_s)$$

formal toplamı olarak tanımlanır.

E, $y^2 = x^3 + ax + b$ denklemi ile verilen bir eliptik eğri ve $f(x, y)$ iki değişkenli bir rasyonel fonksiyon olsun. $P = (x, y) \in E$ olmak üzere $f(P) = f(x, y)$ olarak alınırsa f, E eliptik eğrisi üzerinde tanımlı bir rasyonel fonksiyon olarak düşünülebilir. O halde f fonksiyonunun payını ve paydasını sıfır yapan E eliptik eğrisinin noktaları vardır. Dolayısıyla f fonksiyonunun E eğrisi üzerinde sıfırları ve kutupları vardır. Üstelik sıfır ve kutup yerlerinin katlılıkları da belirtilebilir. Dolayısıyla f fonksiyonu div(f) böleni ile

$$\text{div}(f) = \sum_{P \in E} n_P(P)$$

biçiminde eşleşir. Bu toplamdaki n_P katsayıları tamsayılardır ve sadece sonlu sayıda n_P terimi sıfırdan farklı olduğundan div(f) sonludur. Doğal olarak f fonksiyonunun sıfırları ve kutupları E cisminin tanımlı olduğu cisimden daha büyük bir cisimde bulunabilir.

Daha genel olarak aşağıdaki tanım verilebilir.

3.3.1 Tanım. E bir eliptik eğri, $P \in E$ ve $n_P \in \mathbb{Z}$ olmak üzere

$$D = \sum_{P \in E} n_P(P)$$

biçimindeki herhangi bir formal toplama E üzerinde bir *bölen* denir. Burada sadece sonlu sayıda $P \in E$ için n_P terimi sıfırdan farklı ve diğer tüm n_P terimleri sıfırdır.

Bir D böleninin n_P katsayılarının toplamına D nin *derecesi* denir ve der(D) ile gösterilir, yani

$$\text{der}(D) = \text{der}\,(\sum_{P\in E} n_P(P)) = \sum_{P\in E} n_P$$

dir. Bir D böleninin *toplamı* sum(D) ile gösterilir ve

$$\text{sum}(D) = \text{sum}\,(\sum_{P\in E} n_P(P)) = \sum_{P\in E} n_P P$$

dir.

3.3.2 Örnek 1. E, $y^2 = x^3 + ax + b$ denklemi ile verilen bir eliptik eğri olmak üzere $x^3 + ax + b$ kübik polinomu

$$x^3 + ax + b = (x - \alpha_1)(x - \alpha_2)(x - \alpha_3)$$

olarak çarpanlarına ayrılsın. Bu durumda $P_1 = (\alpha_1, 0)$, $P_2 = (\alpha_2, 0)$ ve $P_3 = (\alpha_3, 0)$ noktaları birbirinden farklıdır ve $2P_1 = 2P_2 = 2P_3 = \boldsymbol{O}$ dur. y fonksiyonunun bu üç noktada sıfırı vardır ve

$$\text{div}(y) = (P_1) + (P_2) + (P_3) - 3(\boldsymbol{O})$$

dur.

2. E bir eliptik eğri ve $P \in E[m]$ noktasının mertebesi m olsun. Bu durumda $mP = \boldsymbol{O}$ olduğundan E eliptik eğrisi üzerinde

$$div(f) = m(P) - m(\boldsymbol{O})$$

olacak biçimde bir f fonksiyonu vardır. Eğer $m = 2$ olarak alınırsa $P = (\alpha, 0) \in E[2]$ olduğundan $f(x) = x - \alpha$ fonksiyonu için

$$div(x - \alpha) = 2(P) - 2(\boldsymbol{O})$$

dur.

3.4. Sonlu Cisimler Üzerinde Tanımlı Eliptik Eğriler

Eliptik eğriler teorisinin kriptoloji uygulamalarında sonlu bir $\mathbb{F}_p$ cismi üzerinde tanımlı eliptik eğriler kullanılır. Eğer bir E eliptik eğrisi sonlu bir $\mathbb{F}_p$ cismi üzerinde tanımlı ise eliptik eğri üzerindeki noktaların x- ve y-koordinatları için sonlu çoklukta olasılık olduğundan $E(\mathbb{F}_p)$ noktalarının kümesinin sonlu bir küme olduğu açıktır. Bir başka ifade ile x-koordinatı için p olasılık varken her x için $y^2 = x^3 + ax + b$ eşitliğinden y için en çok iki olasılık vardır. Böylece sonsuzdaki $\boldsymbol{O}$ noktası ile $E(\mathbb{F}_p)$ grubunun eleman sayısı en çok $2p + 1$ olabilir. Ancak alınan bir noktanın x-koordinatının p modülüne göre bir ikinci dereceden kalan olabilme olasılığı yüzde elli olduğundan $E(\mathbb{F}_p)$ grubundaki noktaların sayısının yaklaşık olarak

$$\#E(\mathbb{F}_p) \approx \frac{1}{2} \cdot 2 \cdot p + 1 = p + 1$$

dir. Hasse'nin aşağıdaki teoremi $\#E(\mathbb{F}_p)$ değeri için bir üst sınır verilmektedir. Bu teoremin ortaya atılmasından yıllar sonra Weil ve Deligne tarafından teorem daha genel hale getirilmiştir.

3.4.1. Teorem (Hasse Teoremi). E, $\mathbb{F}_p$ cismi üzerinde tanımlı bir eliptik eğri olsun. O halde $|t_p| \leq 2\sqrt{p}$ olmak üzere

$$\#E(\mathbb{F}_p) = p + 1 - t_p$$

dir (Silverman 2009).

3.4.2. Tanım. Hasse Teoreminde geçen

$$t_p = p + 1 - \#E(\mathbb{F}_p)$$

değerine $\mathbb{F}_p$ cismi üzerinde tanımlı E eliptik eğrisi için *Frobenius endomorfizminin izi* denir.

3.4.3. Uyarı. Hasse teoremi $\#E(\mathbb{F}_p)$ değeri için bir üst sınır verdiği halde bu değerin hesaplanması için bir yöntem belirtmez. Verilen bir x değeri için $x^3 + ax + b$ değerinin p modülüne göre bir kare olup olmadığını belirlemek $O(p)$ süre alır ve bu kullanışlı değildir. Schoof (1985), $\#E(\mathbb{F}_p)$ sayısını $O((\log p)^6)$ sürede hesaplayan polinom zamanlı bir algoritma geliştirmiştir. Schoof'un algoritması daha sonra Elkies ve Atkin tarafından geliştirildiğinden bu algoritma *SEA algoritması* olarak bilinmektedir.

3.5. Eliptik Eğri Ayrık Logaritma Problemi

Bu kısımda eliptik eğri ayrık logaritma problemi ele alınacaktır. Bunun için ilk olarak $\mathbb{F}_p^*$ grubu için ayrık logaritma problemini hatırlayalım: A kişisi g ve h sayılarını yayınlar ve bu kişinin gizli sayısı x,

$$h \equiv g^x \pmod{p}$$

denkliğinin çözümüdür. Eğer A kişisi g ve h sayılarını $\mathbb{F}_p^*$ grubunun elemanları olarak düşünürse ayrık logaritma problemi, A kişisinin düşmanı olan B kişisinin

$$h \equiv \underbrace{g \cdot \ldots \cdot g}_{x \tan e} \pmod{p}$$

olacak biçimde bir x değerini bulmasını zorunlu hale getirir. Diğer bir ifade ile B kişisi, h yi elde etmek için g nin kaç kere kendisi ile çarpıldığını bulmalıdır.

Benzer şekilde, A kişisi $E(\mathbb{F}_p)$ grubunu kullanarak aynı işlemleri yapabilir. Bunun için A kişisi $E(\mathbb{F}_p)$ grubundan P ve Q noktalarını yayınlar ve

$$Q = \underbrace{P + \ldots + P}_{n \tan e} = nP$$

olacak şekilde bir n tamsayısını gizli tutar. Böylece B kişisinin P noktasını kaç defa kendisiyle toplarsa Q noktasına ulaşacağını bulması gerekir. E eliptik eğrisi üzerindeki

noktaların toplama işlemi daha karışık olduğundan $E(\mathbb{F}_p)$ grubu için ayrık logaritma problemini çözmek oldukça zordur.

3.5.1. Tanım. E, $\mathbb{F}_p$ sonlu cismi üzerinde tanımlı bir eliptik eğri ve $P, Q \in E(\mathbb{F}_p)$ olsun. $Q = nP$ olacak şekildeki bir n tamsayısını bulma problemine *eliptik eğri ayrık logaritma problemi* (EEALP) denir. $\mathbb{F}_p^*$ grubu için ayrık logaritma problemine benzer olarak bu n tamsayısı

$$n = \log_P(Q)$$

ile gösterilir ve n tamsayısına *Q noktasının P noktasına göre eliptik ayrık logaritması* denir.

3.5.2. Uyarı 1. Bu tanımda bazı durumlara dikkat edilmelidir. Örneğin $P, Q \in E(\mathbb{F}_p)$ olmak üzere P noktasının bir katı olacak şekilde bir Q noktası olmayabilir. Bu durumda, $\log_P(Q)$ değeri tanımlı değildir. Ancak, A kişisi herkese açık bir P noktası ve gizli bir n tamsayısı seçip $Q = nP$ değerini hesaplayarak P ve Q noktalarını yayınlamaktadır, böylece $\log_P(Q)$ değeri vardır ve bu değer A kişisinin gizli değeri olmalıdır.

2. Bundan başka, eğer $Q = nP$ olacak biçimde bir n tamsayısı varsa bu özellikte birçok değer bulunabilir. Bunun görmek için ilk önce $sP = \boldsymbol{O}$ olacak şekilde bir pozitif s tamsayısının var olduğunu düşünelim. $E(\mathbb{F}_p)$ sonlu bir grup olduğundan P, $2P$, $3P$, … noktalarının tümü birbirinden farklı olamaz. Böylece $kP = jP$ olacak şekilde $k > j$ tamsayıları vardır ve $s = k - j$ alabiliriz. Bu özellikteki en küçük $s \geq 1$ tamsayısına *P noktasının mertebesi* denir. Böylece eğer, s, P noktasının mertebesi ve $Q = n_0P$ olacak şekilde herhangi bir n_0 tamsayısı varsa $Q = nP$ eşitliğinin çözümleri, $i \in \mathbb{Z}$ olmak üzere $n = n_0 + is$ tamsayılarıdır. O halde $\log_P(Q)$ değeri $\mathbb{Z}/s\mathbb{Z}$ nin bir elemanıdır, yani s, P noktasının mertebesi olmak üzere $\log_P(Q)$, s modülüne göre bir tamsayıdır. Genellikle $\log_P(Q) = n_0$ olarak alınır. $\log_P(Q)$ değeri $\mathbb{Z}/s\mathbb{Z}$ nin bir elemanı olduğundan eliptik ayrık logaritması her $Q_1, Q_2 \in E(\mathbb{F}_p)$ için

$$\log_P(Q_1 + Q_2) = \log_P(Q_1) + \log_P(Q_2) \qquad (3.\ 4)$$

dir. Böylece (3. 4) eşitliğinden $E(\mathbb{F}_p)$ grubu için EEALPnin, klasik logaritma özelliğine ve $\mathbb{F}_p^*$ için ayrık logaritma özelliğine benzediği görülmektedir. Bundan başka (3. 4) eşitliği kullanılarak $\log_P$ fonksiyonunun $E(\mathbb{F}_p)$ grubundan $\mathbb{Z}/s\mathbb{Z}$ grubuna bir grup homomorfizm olduğu kolayca görülebilir.

3.5.3. Örnek. $\mathbb{F}_{73}$ sonlu cismi üzerinde tanımlı $E : y^2 = x^3 + 8x + 7$ eliptik eğrisi dikkate alınırsa $P = (32, 53)$, $Q = (39, 17) \in E(\mathbb{F}_{73})$ için $Q = 11P$ ve böylece $\log_P Q = 11$ dir. Benzer şekilde $R = (35, 47)$, $S = (58, 4) \in E(\mathbb{F}_{73})$ noktaları için $R = 37P$ ve $S = 28P$ olduğundan

$$\log_P(R) = 37 \quad \text{ve} \quad \log_P(S) = 28$$

dir. Diğer yandan $\#E(\mathbb{F}_{73}) = 82$ olduğu halde P noktası için $41P = \boldsymbol{O}$, yani P noktasının mertebesi $82/2 = 41$ dir. Dolayısıyla $E(\mathbb{F}_{73})$ grubundaki noktaların yarısı P noktasının bir katıdır. Örneğin, $(20, 65) \in E(\mathbb{F}_{73})$ noktası P noktasının bir katına eşit değildir (Hoffstein ve ark. 2008).

3.6. İkiye Katlama ve Toplama Algoritması

$E(\mathbb{F}_p)$ grubundaki P ve $Q = nP$ noktalarından n değerini elde etmek, yani EEALPni çözmek zordur. Bununla birlikte

$$\mathbb{Z} \to E(\mathbb{F}_p)\ ,\ n \to nP$$

fonksiyonunu kriptolojide kullanmak için bilinen n ve P değerlerinden nP değeri verimli bir biçimde elde edilmelidir. Eğer n sayısı büyük bir sayı ise P, $2P$, $3P$, … gibi değerleri hesaplayarak nP değerini bulmak zaman alacaktır. Dolayısıyla nP değerini hesaplamak için daha önce Diffie-Hellman anahtar değişimi, ElGamal ve RSA açık anahtar kriptosistemleri için $a^n \pmod N$ kuvvetlerinin hesaplanması için uygulanan hızlı

kuvvet alma algoritmasına benzer bir yöntem kullanılabilir. Bunun için ilk olarak n sayısı $n_0, n_1, n_2, \ldots, n_r \in \{0, 1\}$ olmak üzere

$$n = n_0 + n_1 \cdot 2 + n_2 \cdot 4 + n_3 \cdot 8 + \cdots + n_r \cdot 2^r$$

şeklinde yazılır ve $n_r = 1$ olduğu varsayılarak

$$Q_0 = P,\ Q_1 = 2Q_0,\ \ Q_2 = 2Q_1,\ \ldots,\ Q_r = 2Q_{r-1}$$

değerleri hesaplanır. Bu eşitliklerden görülebileceği gibi $Q_i = 2Q_{i-1}$ olduğundan

$$Q_i = 2^i P$$

olduğu elde edilir. Bu noktalar P noktasının 2 nin bir kuvvetinin katıdır ve bu noktaların hesaplanması en çok r tane ikiye katlama gerektirir. Son olarak nP noktası en çok r tane toplama işlemi ile

$$nP = n_0 Q_0 + n_1 Q_1 + \cdots + n_r Q_r$$

biçiminde hesaplanır.

$E(\mathbb{F}_p)$ grubundaki iki noktanın toplama işlemi bir nokta işlemi olarak alınırsa nP değerini hesaplamak için $E(\mathbb{F}_p)$ grubunda en çok $2r$ tane nokta işlemi yapılmış olur. $n \geq 2^r$ olduğundan nP değerini hesaplamak için $2\log_2(n)$ tane nokta işleminden daha fazla işleme gerek yoktur. Üstelik bu yöntemle n sayısının çok büyük değerleri için de nP değerini hesaplamak mümkün olur.

Aşağıda bu yöntem için bir algoritma verilecektir.

3.6.1. Algoritma (İkiye Katlama ve Toplama Algoritması).

Input. $P \in E(\mathbb{F}_p)$ noktası ve $n \geq 1$ tamsayısı.

1. $Q = P$ ve $R = \boldsymbol{O}$ al.

2. $n > 0$ iken

3. $n \equiv 1 \pmod 2$ ise, $R = R + Q$ al.

4. $Q = 2Q$ ve $n = \lfloor n/2 \rfloor$ al.

5. $n > 0$ ise 2. adıma git.

6. $nP = R$ noktasına dön (Hoffstein ve ark. 2008).

3.6.2. Örnek. Yukarıda verilen İkiye Katlama ve Toplama algoritmasını kullanarak $E : y^2 = x^3 - 21x + 23$ eliptik eğrisi ve $n = 187$**,** $p = 1531$, $P = (5, 434)$ için $E(\mathbb{F}_p)$ grubunda nP değerini hesaplanmak istenirse n sayısı ikinin kuvvetleri biçiminde

$$n = 187 = 1 + 2 + 2^3 + 2^4 + 2^5 + 2^7$$

olarak yazılabilir. Böylece n sayısının yazılışından da anlaşılacağı gibi yedi ikiye katlama ve beş toplama işlemi yapıldığı görülür. Sonuç olarak $187P = (935, 866)$ olarak bulunur.

3.6.3. Uyarı. nP değerini hesaplamak için gereken süreyi daha da azaltmanın bir yöntemi n sayısını 2 nin kuvvetlerinin farkı ve toplamını kullanarak yazmaktır. Böylece nP değerini hesaplamak için daha az nokta kullanılır. Weierstrass formda verilen bir eliptik eğri üzerindeki herhangi bir nokta için $-(x, y) = (x, -y)$ olduğundan iki noktanın farkı**,** toplamları kadar kolay bulunur. $\mathbb{F}_p^*$ grubunda bir a elemanı için a^{-1} değerinin hesaplanması herhangi iki elemanın çarpımı işleminden daha fazla zaman aldığından $E(\mathbb{F}_p)$ grubunda yapılan yukarıdaki işlemler $\mathbb{F}_p^*$ grubundan farklıdır.

Yukarıdaki örnekte, $187 = 1 + 2 + 2^3 + 2^4 + 2^5 + 2^7$ olarak yazıldığından $187P$ değerini hesaplamak için 7 ikiye katlama ve 5 toplama işlemi olmak üzere toplam 12 nokta işlemi yapılmıştır. Ancak n sayısı

$$187 = -1 - 2^2 - 2^6 + 2^8$$

şeklinde yazılırsa

$$187P = -P - 2^2P - 2^6P + 2^8P$$

yi hesaplamak için 8 ikiye katlama ve 3 toplama işlemi olmak üzere toplam 11 nokta işlemi yapılmış olur. Bir n sayısının 2'nin pozitif ve negatif kuvvetlerinin bir toplamı olarak yazılmasına n sayısının bir *üçlü açılımı* denir.

Şimdi n sayısının büyük bir sayı olduğunu varsayalım ve $k = \lfloor \log n \rfloor + 1$ olarak alalım. n sayısı $2^k - 1$ biçiminde yazılabiliyorsa

$$2^k - 1 = 1 + 2 + 2^2 + \cdots + 2^{k-1}$$

olduğundan n sayısının bir ikili açılımını kullanarak nP değerinin hesaplanması k tane ikiye katlama ve k tane toplama işlemi olmak üzere toplam $2k$ tane nokta işlemini gerektirir. Eğer n sayısının yazılışında üçlü açılım kullanılırsa Önerme 3.6.4'te görüleceği gibi nP değerini hesaplamak için $k + 1$ tane ikiye katlama ve $\frac{1}{2}k$ tane toplama işlemi olmak üzere toplam $\frac{3}{2}k + 1$ nokta işleminden daha fazla nokta işlemi gerekmez.

Şimdi genel duruma bakalım. Rastgele bir sayının ikili açılımındaki 0'ların ve 1'lerin sayısı yaklaşık olarak aynıdır. Dolayısıyla n sayısının ikili açılımı kullanarak nP değerini hesaplamak k tane ikiye katlama ve $\frac{1}{2}k$ tane toplama işlemi olmak üzere toplam $\frac{3}{2}k$ adım gerekir. Eğer 2'nin kuvvetlerinin toplamları ve farkları alınırsa n sayısının ikili açılımında terimlerin $\frac{2}{3}$'ü sıfır olduğundan nP değerini hesaplarken $k + 1$ tane ikiye katlama ve k tane toplama işlemi olmak üzere toplam $\frac{4}{3}k + 1$ adım gerekir.

3.6.4. Önerme. n bir pozitif tamsayı ve $k = \lfloor \log n \rfloor + 1$, yani $2^k > n$ olsun. $u_0, u_1, \ldots, u_k \in \{-1, 0, 1\}$ ve u_i değerlerinin en çok $\frac{1}{2}k$ tanesi sıfırdan farklı olmak üzere

$$n = u_0 + u_1 \cdot 2 + u_2 \cdot 4 + u_3 \cdot 8 + \cdots + u_k \cdot 2^k \qquad (3.5)$$

olarak yazılabilir (Hoffstein ve ark. 2008).

İspat. Önermenin ispatı aslında n sayısını istenen biçimde yazmak için kullanılan bir algoritmadır. n sayısı, $n_0, \dots, n_{k-1} \in \{0, 1\}$ olmak üzere

$$n = n_0 + n_1 \cdot 2 + n_2 \cdot 4 + n_3 \cdot 8 + \cdots + n_{k-1} \cdot 2^{k-1}$$

olarak ikili biçimde yazılabilir. Ardışık sıfırdan farklı n_i katsayılarını bulmak istiyoruz. Belli bir $t \geq 2$ için

$$n_s = n_{s+1} = \cdots = n_{s+t-1} = 1 \text{ ve } n_{s+t} = 0$$

olduğunu varsayalım, diğer bir ifade ile n sayısının ikili açılımında

$$2^s + 2^{s+1} + \cdots + 2^{s+t-1} + 0 \cdot 2^{s+t} \tag{3. 6}$$

ifadesinin olduğunu düşünelim. Bu ifade

$$2^s + 2^{s+1} + \cdots + 2^{s+t-1} + 0 \cdot 2^{s+t} = 2^s (1 + 2 + 4 + 2^{t-1}) = 2^s (2^t - 1)$$

şeklinde yazılabildiğinden (3. 6) ifadesi

$$-2^s + 2^{s+t}$$

olarak yazılabilir. Bu işleme devam edilirse ardışık iki u_i değeri sıfırdan farklı olmayacak şekilde n sayısının (3. 5) deki gibi bir yazılımı elde edilmiş olur.

3.7. EEALP Ne Kadar Zordur?

Kısım 2.4. de ele alınan çarpışma algoritmaları herhangi bir gruba kolayca uygulanabilir. Dolayısıyla bir eliptik eğrinin üzerindeki noktaların grubu $E(\mathbb{F}_p)$ için böyle bir algoritma verilebilir. $Q = nP$ eşitliğini çözmek için bir B kişisi 1 ve p asal sayısı arasında rastgele $j_1, j_2, \dots, j_r$ ve $k_1, k_2, \dots, k_r$ tamsayıları seçer ve aşağıdaki gibi iki liste oluşturur.

1. Liste: $j_1P, j_2P, j_3P, \dots, j_rP$,
2. Liste: $k_1P + Q, k_2P + Q, k_3P + Q, \dots, k_rP + Q$.

B kişisi iki liste arasında $j_uP = k_vP + Q$ eşlemesini bulursa $Q = (j_u - k_v)P$ olduğundan çözümü elde eder.

Bu çarpışma algoritması iki liste için fazla depolama alanı gerektirir. Bununla birlikte, Kısım 2.5. te ele alınan Pollard'ın ρ algoritması daha az depolama alanına sahiptir ve benzer bir çalışma süresiyle buradaki duruma uygulanabilir. Her halde $E(\mathbb{F}_p)$ grubu için EEALPni $O(\sqrt{p})$ adımda çözen algoritmalar vardır.

$\mathbb{F}_p^*$ grubu için ayrık logaritma problemini çözmenin daha hızlı yollarının var olduğunu biliyoruz. Özellikle, daha önce ele alınan indeks hesabı yöntemi alt üstel çalışma süresine sahiptir, yani bu yöntemin çalışma süresi her $\varepsilon > 0$ için $O(p^{\varepsilon})$ dir. Eliptik eğrilerin kriptolojide kullanılmasının temel nedeni, EEALP için indeks hesabı yöntemlerinin bulunmamasıdır. Gerçekten de EEALPni $O(\sqrt{p})$ adımdan daha az sürede çözmek için bilinen genel algoritmalar yoktur. EEALPni çözmek için bilinen en hızlı algoritmalar herhangi bir gruptaki ALPni için aynı derecede çalışan algoritmalardan daha iyi değildir. Böylece EEALP, ALPnden çok daha zordur. Bununla birlikte $\mathbb{F}_p^*$ grubu için ALPnin kolay çözülebildiği p asal sayıları vardır. Örneğin, $p - 1$ sayısı küçük asal sayıların çarpımı şeklinde yazılabiliyorsa, Pohlig-Hellman algoritması $\mathbb{F}_p^*$ grubu için ALPni hızlı bir şekilde çözüme ulaştırır.

3.8. Eliptik Eğri Kriptolojisi

Bu kısımda eliptik eğrilerin kriptolojiye nasıl uygulandığını ele alacağız. $E(\mathbb{F}_p)$ grubu için ayrık logaritma problemi ile anahtar değişimini güvenli bir şekilde sağlayan en basit uygulama Diffie-Hellman anahtar değişimi algoritmasıdır.

3.8.1. Eliptik Diffie-Hellman Anahtar Değişimi

Bu kısımda daha önce $\mathbb{F}_p$ grubunda ele alınan Diffie-Hellman anahtar değişimi yöntemi, bir E eliptik eğrisi üzerindeki noktaların grubu $E(\mathbb{F}_p)$ üzerinde ele alınacaktır. Buna göre, A ve B kişileri bir $E(\mathbb{F}_p)$ grubu ve $P \in E(\mathbb{F}_p)$ noktasını kullanırlar.

- İlk olarak A kişisi gizli bir n_A tamsayısı seçer ve B kişisi de gizli bir n_B tamsayısı seçer.
- İkinci olarak A kişisi $Q_A = n_A P$ değerini ve B kişisi de $Q_B = n_B P$ değerini hesaplar.
- Sonraki adımda A kişisi Q_A değerini B kişisine, B kişisi de Q_B değerini A kişisine gönderir. Böylece A kişisi gizli n_A değerini kullanarak $n_A Q_B$ yi ve B kişisi de n_B değerini kullanarak $n_B Q_A$ yı hesaplar ve paylaştıkları gizli değerlerle birlikte

  $$n_A Q_B = (n_A n_B)P = n_B$$

  gizli değerine ulaşırlar. Bu gizli değeri A ve B kişileri simetrik şifreyle iletişim kurma k için anahtar olarak kullanabilirler.

3.8.2. Örnek. A ve B kişileri $p = 3851$ asal sayısı, E: $y^2 = x^3 + 324x + 1287$ eliptik eğrisi ve $P = (920, 303) \in E(\mathbb{F}_{3851})$ noktası ile eliptik Diffie-Hellman anahtar değişimi yöntemini kullanmaya karar verirler. A kişisi $n_A = 1194$ ve B kişisi $n_B = 1759$ gizli değerlerini seçerler ve daha sonra A kişisi

$$Q_A = 1194P = (2067, 2178) \in E(\mathbb{F}_{3851})$$

değerini B kişisi

$$Q_B = 1759P = (3684, 3125) \in E(\mathbb{F}_{3851})$$

değerini hesaplar. A kişisi Q_A değerini B kişisine, B kişisi de Q_B değerini A kişisine gönderir. Sonuç olarak, A ve B kişileri

$$n_A Q_B = 1194(3684, 3125) = (3347, 1242) = n_B Q_A = 1759(2067, 3178) \in E(\mathbb{F}_{3851})$$

hesaplamalarını yaparak (3347, 1242) gizli noktasını paylaşmış olurlar. Uyarı 3.8.4 te açıklanacağı gibi A ve B kişileri gizli değerin y koordinatını kullanmayıp sadece x = 3347 gizli değerini kullanırlar. Eğer bir C kişisi A ve B kişilerinin gizli değerini çözmek için

$$nP = Q_A$$

EEALPni çözebilirse n_A değerini bildiğinden $n_A Q_B$ değerini hesaplayabilir (J. Hoffstein ve ark. 2008).

3.8.3. Tanım. E, $\mathbb{F}_p$ sonlu cismi üzerinde tanımlı bir eliptik eğri ve $P \in E(\mathbb{F}_p)$ olsun. Bilinen n_1P ve n_2P değerlerinden n_1n_2P değerini hesaplama problemine *eliptik eğri Diffie-Hellman problemi* denir.

3.8.4. Uyarı 1. Eliptik Diffie-Hellman anahtar değişimi, A ve B kişilerinin bir eliptik eğri üzerinde nokta alışverişi yapmasını gerektirir. $E(\mathbb{F}_p)$ grubundaki bir Q noktası, x_Q ve y_Q $\mathbb{F}_p$ sonlu cisminin elemanları olmak üzere $Q = (x_Q, y_Q)$ biçiminde olduğundan A kişisi B kişisine $\mathbb{F}_p$ de iki sayı göndermek zorundadır. Bununla birlikte, bu iki sayı arasında $\mathbb{F}_p$ cisminde

$$y_Q^2 = x_Q^3 + ax_Q + b$$

biçiminde bir ilişki olduğundan bu iki sayı p modülüne göre iki keyfi sayı kadar bilgi vermez. Dikkat edilirse bir C kişisi a ve b değerlerini bilmektedir ve dolayısıyla C kişisi x_Q nun doğru değerini tahmin edebilirse y_Q için iki muhtemel değer olduğundan C kişisi için y_Q değerini hesaplamak zor değildir.

2. Dolayısıyla A kişisinin Q_A noktasının iki koordinatını B kişisine göndermesine gerek yoktur. O halde A kişisi B kişisine sadece Q_A noktasının x-koordinatını gönderir. Böylece B kişisi de mümkün olan iki y koordinatından birini hesaplar ve kullanır. Eğer B kişisi doğru y değerini seçerse Q_A değerini kullanır. Eğer B kişisi yanlış y değerini seçerse $-Q_A$ değerini kullanmış olur. Her iki durumda da B kişisi

$$\pm n_B Q_A = \pm (n_A n_B)P$$

değerlerinden birini hesaplar. Benzer şekilde A kişisi de $\pm(n_A n_B)P$ değerlerinden birini hesaplar. Böylece A ve B kişileri hangi y değerini kullandıklarına bakmaksızın x-koordinatı aynı olduğundan paylaşılan gizli değerleri olarak x-koordinatını kullanırlar.

Şimdi eliptik Diffie-Hellman anahtar değişimine bir örnek verelim. Bunun için aşağıdaki uyarıya ihtiyaç vardır.

3.8.5. Uyarı. p, $p \equiv 3 \pmod 4$ özelliğinde bir asal sayı ve a sayısı $x^2 \equiv a \pmod p$ denkliğinin bir çözümü olacak şekilde bir tamsayı olsun. O halde

$$b \equiv a^{(p+1)/4} \pmod p$$

bir çözümdür, yani $b^2 \equiv a \pmod p$ denkliğini gerçekler.

3.8.6. Örnek. A ve B kişileri Örnek 3.8.2'deki $p = 3851$ asal sayısı, E: $y^2 = x^3 + 324x + 1287$ eliptik eğrisi ve $P = (920, 303) \in E(\mathbb{F}_{3851})$ noktasını kullanarak birbirlerine daha az bit göndererek başka bir gizli değer alışverişi yapmaya karar verirler. A ve B kişileri yeni gizli değerlerini $n_A = 2489$, $n_B = 2286$ olarak seçerler ve A kişisi

$$Q_A = n_A P = 2489(920, 303) = (593, 719) \in E(\mathbb{F}_{3851})$$

değerini, B kişisi de

$$Q_B = n_B P = 2286(920, 303) = (3681, 612) \in E(\mathbb{F}_{3851})$$

değerini hesaplar. Bununla birlikte, noktanın iki koordinatını göndermek yerine A kişisi B kişisine sadece $x_A = 593$ değerini ve B kişisi de A kişisine sadece $x_B = 3681$ değerini gönderir. Böylece, A kişisi E eliptik eğrisinin denkleminde $x_B = 3681$ değerini yazarsa,

$$y_B^2 = x_B^3 + 324\, x_B + 1287 = 3681^3 + 324 \cdot 3681 + 1287 = 997$$

değerini bulur ve A kişisinin 3851 modülüne göre 997 sayısının bir karekökünü hesaplaması gerekir. Böylece A kişisi $p \equiv 3 \pmod 4$ olduğundan yukarıdaki uyarıyı dikkate alarak

$$y_B = 997^{(3851+1)/4} = 997^{963} \equiv 612 \pmod{3851}$$

değerini hesaplar. Böylece B kişisinin kullandığı $Q_B = (x_B, y_B) = (3681, 612)$ noktasını elde eder ve $n_A Q_B = 2489(3681, 612) = (509, 1108)$ değerini hesaplar.

Benzer şekilde, B kişisi $x_A = 593$ değerini E eliptik eğri denkleminde yerine yazar ve karekök alırsa,

$$y_A^2 = x_A^3 + 324x_A + 1287 = 593^3 + 324 \cdot 593 + 1287 = 927$$

$$y_A = 927^{(3851+1)/4} = 927^{963} \equiv 3132 \pmod{3851}$$

olarak elde eder. B kişisi, A kişisinin Q_A noktasını kullanmayıp $Q_A' = (593, 3132)$ noktasını kullanır ve

$$n_B Q_A' = 2286(593, 3132) = (509, 2743)$$

değerini hesaplar. B kişisinin yaptığı bu hesaplamada $E(\mathbb{F}_p)$ deki noktanın negatifi kullandığı halde bu noktaların x-koordinatı aynı olduğundan A ve B kişilerinin paylaştığı gizli değer $x = 509$ dur (Hoffstein ve ark. 2008).

3.9. Eliptik ElGamal Açık Anahtar Kriptosistemi

Bu kısımda daha önce Kısım 2.4. te ele alınan ElGamal açık anahtar kriptosistemine benzer bir kriptosistemin eliptik eğrilere uygulamasını ele alınacaktır. Eliptik ElGamal açık anahtar kriptosistemine göre, A ve B kişileri belli bir p asal sayısı, E eliptik eğrisi ve $P \in E(\mathbb{F}_p)$ noktasını kullanır. A kişisi gizli bir n_A değeri (özel anahtar) seçer ve $E(\mathbb{F}_p)$ grubunda $Q_A = n_A P$ değerini hesaplayıp Q_A değerini açık anahtarı olarak yayınlar. B kişisi bir $M \in E(\mathbb{F}_p)$ noktasını düz metni olarak alır ve rastgele bir k tamsayısı seçip A kişisinin Q_A değerini kullanarak

$$c_1 = kP, \quad c_2 = M + k\,Q_A \in E(\mathbb{F}_p)$$

değerlerini hesaplar. Daha sonra B kişisi (c_1, c_2) noktalarını A kişisine gönderir. Böylece A kişisi

$$c_2 - n_A c_1 = (M + kQ_A) - n_A(kP) = M + k\,(n_A P) - n_A(kP) = M$$

hesaplamasını yaparak M düz metnine ulaşır.

3.9.1. Uyarı 1. Eliptik ElGamal açık anahtar kriptosisteminde düz metin mesajlarını $E(\mathbb{F}_p)$ noktaları ile bağlamanın açık bir yolu yoktur.

2. $\mathbb{F}_p$ cismi kullanılırsa ElGamal 2'ye 1 mesaj kriptosistemi olduğu halde eliptik ElGamal 4'e 1 mesaj kriptosistemidir. Eliptik ElGamal kriptosisteminde 4'e karşılık 1 mesaj elde edilmesinin nedeni, M düz metninin $E(\mathbb{F}_p)$ de tek bir nokta olmasıdır. Hasse Teoremi'ne göre, $E(\mathbb{F}_p)$ de yaklaşık olarak p farklı nokta olması p farklı düz metin olabileceğini gösterir. Bununla birlikte, $E(\mathbb{F}_p)$ deki her nokta iki koordinattan oluştuğundan (c_1, c_2) şifreli metni p modülüne göre dört sayıdan oluşur.

3. Eliptik Diffie-Hellman anahtar değişiminde olduğu gibi c_1 ve c_2 nin sadece x-koordinatlarını karşı tarafa gönderilebilir. Ancak, A kişisinin $c_2 - n_A c_1$ değerini hesaplaması gerektiğinden A kişisinin c_1 ve c_2 değerlerinin hem x- hem de y-koordinatının doğru değerlerine ihtiyacı vardır. Ayrıca bir noktanın x-koordinatı ile işarete bağlı olarak y-koordinatı belirlenebilir. Bu nedenle B kişisi, A kişisine ekstra bit yollamış olur, örneğin,

$$\text{Ekstra bit} = \begin{cases} 0, & 0 \le y < \dfrac{1}{2}p \\ 1, & \dfrac{1}{2}p < y < p \end{cases}$$

dir. Böylece B kişisi sadece c_1 ve c_2 değerlerinin x-koordinatlarını ve ekstra iki bit göndermelidir.

3.10. Weil Eşleştirmesi

Aşağıda eliptik eğri kriptolojisinde ve özellikle MOV algoritmasında kullanılan Weil eşleştirmesi kavramı ve özellikleri üzerinde durulacaktır. İlk olarak Weil eşleştirmesi tanımı ile başlayalım.

3.10.1. Tanım. E, bir $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri, m bir pozitif tamsayı ve $\mathbb{F}$ cisminin karakteristiği ile m aralarında asal olmak üzere $P, Q \in E[m]$ olsun. f_P ve f_Q, E eliptik eğrisi üzerinde bölenleri

$$\text{div}(f_P) = m[P] - m[O] \qquad \text{ve} \qquad \text{div}(f_Q) = m[Q] - m[O]$$

olan rasyonel fonksiyonlar olmak üzere $S \in E(\mathbb{F})$ eliptik eğrisi üzerinde ve $S \notin \{O, P, -Q, P - Q\}$ özelliğinde herhangi bir nokta olsun. Bu durumda $\boldsymbol{\mu}_m$, birimin m. ilkel köklerinin grubu olmak üzere

$$e_m : E[m] \times E[m] \to \boldsymbol{\mu}_m,\ e_m(P, Q) = \frac{f_P(Q+S)}{f_P(S)} \Big/ \frac{f_Q(P-S)}{f_Q(-S)}$$

biçiminde verilen e_m eşleştirmesine *P ve Q noktalarının Weil eşleştirmesi* (*Weil pairing*) denir. Burada $e_m(P, Q)$ değeri f_P, f_Q fonksiyonları ve S noktasının seçiminden bağımsızdır.

3.10.2. Uyarı 1. Her $P, Q \in E[m]$ için e_m Weil eşleştirmesinin

$$e_m(P, Q)^m = 1$$

eşitliğini gerçeklediği açıktır. Diğer bir ifade ile $e_m(P, Q)$ birimin m. köküdür.

2. Hatırlanacağı gibi $E[m]$ grubu $\mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$ grubuna izomorftur. Dolayısıyla $E[m]$, rankı 2 olan serbest $\mathbb{Z}/m\mathbb{Z}$-modüldür. O halde $\{P_1, P_2\}$, $E[m]$ için bir baz olmak üzere determinant dönüşümü kullanılarak

$$\det: E[m] \times E[m] \to \mathbb{Z}/m\mathbb{Z},\ \det(aP_1 + bP_2,\ cP_1 + dP_2) = ad - bc$$

biçiminde bir alterne bilineer eşleştirme tanımlanabilir. $ad - bc$ değerinin bazın seçiminden bağımsız olduğu açıktır. Üstelik Weil eşleştirmesi ile determinant eşleştirmesi birbirleri ile yakından ilgilidir.

3. $E[m]$ üzerindeki determinant eşleştirmesi bir Galois değişmezi değildir. Diğer bir ifade ile $P, Q \in E[m]$ ve $\sigma \in G(\overline{K}/K)$ ise $\det(\sigma(P), \sigma(Q))$ değeri $\sigma(\det(P, Q))$ olmak zorunda değildir. Aşağıdaki teoremde e_m Weil eşleştirmesinin $E[m]$ üzerinde Galois değişmezliği özelliğini gerçeklediği görülecektir. Buna göre ζ, birimin m. ilkel kökü olmak üzere $\zeta = e_m(P, Q)$ olarak alınırsa

$$\zeta = e_m(P, Q) = e_m(\sigma(P), \sigma(Q)) = \sigma(e_m(P, Q)) = \sigma(\zeta)$$

dir.

Aşağıdaki teoremde Weil eşleştirmesinin özellikleri belirtilmektedir.

3.10.3. Teorem. E, bir $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri, m bir pozitif tamsayı olmak üzere $\mathbb{F}$ cisminin karakteristiği m sayısını bölmesin. $\boldsymbol{\mu}_m$, birimin m. ilkel köklerinin grubu olsun. Bu durumda

$$e_m : E[m] \times E[m] \to \boldsymbol{\mu}_m$$

Weil eşleştirmesi aşağıdaki özellikleri gerçekler.

i) e_m eşleştirmesi bilineerdir, yani her $P, P_1, P_2, Q, Q_1, Q_2 \in E[m]$ için

$$e_m(P_1 + P_2, Q) = e_m(P_1, Q)\, e_m(P_2, Q),$$
$$e_m(P, Q_1 + Q_2) = e_m(P, Q_1)\, e_m(P, Q_2)$$

dir.

ii) e_m eşleştirmesi alternedir, yani her $P \in E[m]$ için

$$e_m(P, P) = 1$$

dir. Özel olarak her $P, Q \in E[m]$ için

$$e_m(P, Q) = e_m(Q, P)^{-1}$$

dir.

iii) e_m eşleştirmesi dejenere değildir, yani her $Q \in E[m]$ için

$$e_m(P, Q) = 1 \text{ ise } Q = \boldsymbol{O}$$

dur.

iv) e_m eşleştirmesi Galois değişmezidir, yani $G(\overline{K}/K)$, $\overline{K}$ cisminin K cismi üzerinde Galois grubu olmak üzere her $\sigma \in G(\overline{K}/K)$ için

$$e_m(\sigma P, \sigma Q) = \sigma(e_m(P, Q))$$

dir.

v) e_m eşleştirmesi uyumludur, yani her $P \in E[mm']$ ve $Q \in E[m]$ için

$$e_{mm'}(P, Q) = e_m(m'P, Q)$$

dur (Silverman 2009).

3.10.4. Örnek. Şimdi Weil eşleştirmesi tanımını kullanarak

$$E : y^2 = x^3 + ax + b = (x - \alpha_1)(x - \alpha_2)(x - \alpha_3)$$

eliptik eğrisi için e_2 değerini hesaplayalım. Dikkat edilirse yukarıdaki eşitliğin sol tarafında x^2 li terim olmadığından $\alpha_1 + \alpha_2 + \alpha_3 = 0$ dır. Diğer yandan $P_1 = (\alpha_1, 0)$, $P_2 = (\alpha_2, 0)$ ve $P_3 = (\alpha_3, 0)$ noktalarının mertebesi 2 olduğundan

$$\text{div}(x - \alpha_i) = 2[P_i] - 2[O]$$

dur.

$e_2(P_1, P_2)$ değerini hesaplamak için E eliptik eğrisi üzerinde keyfi bir $S = (x, y)$ noktası alalım. Eliptik eğriler üzerindeki toplama işlemi formülleri kullanılırsa $P_1 - S$ noktasının x-koordinatı bulunabilir. Buna göre

$$x(P_1 - S) = \left(\frac{-y}{x-\alpha_1}\right) - x - \alpha_1 = \frac{y^2 - (x-\alpha_1)^2(x+\alpha_1)}{(x-\alpha_1)^2}$$

dir. Eğer $y^2 = (x - \alpha_1)(x - \alpha_2)(x - \alpha_3)$ olduğu dikkate alınırsa

$$x(P_1 - S) = \frac{(x-\alpha_1)(x-\alpha_2)(x-\alpha_3) - (x-\alpha_1)^2(x+\alpha_1)}{(x-\alpha_1)^2} = \frac{(-\alpha_2-\alpha_3)x + \alpha_2\alpha_3 + \alpha_1^2}{x-\alpha_1}$$

dır. Son olarak $\alpha_1 + \alpha_2 + \alpha_3 = 0$ olduğundan

$$x(P_1 - S) = \frac{\alpha_1 x + \alpha_2\alpha_3 + \alpha_1^2}{x-\alpha_1}$$

olarak bulunur. Benzer biçimde

$$x(P_2 + S) = \frac{\alpha_2 x + \alpha_1\alpha_3 + \alpha_2^2}{x-\alpha_2}$$

olarak elde edilir.

Şimdi $e_2(P_1, P_2)$ değerini hesaplayalım. Bunun için $f_{P_i} = x - \alpha_i$ rasyonel fonksiyonlarını kullanalım ve P_1 ve P_2 noktalarının $E[2]$ de sıfırdan ve birbirinden farklı noktalar olduğu varsayalım. O halde e_m eşleştirmesinin tanımından

$$e_2(P_1, P_2) = \frac{f_{P_1}(P_2 + S)}{f_{P_1}(S)} \Big/ \frac{f_{P_2}(P_1 - S)}{f_{P_2}(-S)}$$

$$= \frac{X(P_2 + S) - a_1}{X(S) - a_1} \Big/ \frac{X(P_1 - S) - a_2}{X(-S) - a_2}$$

$$= \frac{\dfrac{\alpha_2 x + \alpha_1\alpha_3 + \alpha_2^2}{x-\alpha_2} - \alpha_1}{x-\alpha_1} \Big/ \frac{\dfrac{\alpha_1 x + \alpha_2\alpha_3 + \alpha_1^2}{x-\alpha_1} - \alpha_2}{x-\alpha_2}$$

eşitliği elde edilir. Bu eşitlik yeniden düzenlenir ve $\alpha_1 + \alpha_2 + \alpha_3 = 0$ olduğu kullanılırsa

$$e_2(P_1, P_2) = \frac{(\alpha_2 - \alpha_1)x + \alpha_2^2 - \alpha_1^2}{(\alpha_1 - \alpha_2)x + \alpha_1^2 - \alpha_2^2} = -1$$

olarak bulunur (Hoffstein ve ark. 2008).

3.11. Gömme Derecesi ve MOV Algoritması

Weil eşleştirmesinin asal kuvvet mertebeli $\mathbb{F}_{p^k}$ cisimlerinde çalışıldığı birçok uygulaması vardır. Bu kısımda bu uygulamalardan biri olan MOV algoritmasından bahsedilecektir. Bu algoritma $E(\mathbb{F}_p)$ grubu üzerinde tanımlı olan EEALPni $\mathbb{F}_{p^k}$ cismindeki ALPne indirgediğinden etkili bir algoritmadır. MOV algoritmasını daha iyi anlayabilmek için aşağıdaki tanıma ihtiyacımız vardır.

3.11.1. Tanım. E, $\mathbb{F}_p$ sonlu cismi üzerinde tanımlı bir eliptik eğri, $m \geq 1$ bir tamsayı olmak üzere $p \nmid m$ olsun.

$$E(\mathbb{F}_{p^k})[m] \cong \mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$$

olacak şekildeki en küçük k değerine *E eliptik eğrisinin m ye göre gömme derecesi* denir.

Aşağıdaki önermede bir E eliptik eğrisinin gömme derecesinin nasıl belirlendiği görülmektedir.

3.11.2. Önerme. E, $\mathbb{F}_p$ sonlu cismi üzerinde tanımlı bir eliptik eğri ve $l \neq p$ bir asal sayı olsun. $E(\mathbb{F}_p)$ grubunda l mertebeli bir nokta olduğunu varsayalım. Bu durumda, E eliptik eğrisinin l asal sayısına göre gömme derecesi aşağıdakilerden biridir:

a) $l < \sqrt{p} + 1$ ise E eliptik eğrisinin gömme derecesi 1'dir.

b) $p \equiv 1 \pmod{l}$ ise gömme derecesi l'dir.

c) $p \not\equiv 1 \pmod{l}$ ise gömme derecesi $p^k \equiv 1 \pmod{l}$ olacak şekildeki en küçük $k \geq 2$ değeridir (Hoffstein ve ark. 2008).

E eliptik eğrisinin k gömme derecesi, $E(\mathbb{F}_p)$ üzerindeki EEALPni $\mathbb{F}_{p^k}$ sonlu cismindeki ALPne indirgediğinden oldukça önemlidir. Buna göre, E, $\mathbb{F}_p$ sonlu cismi üzerinde tanımlı bir eliptik eğri ve $l > \sqrt{p} + 1$ bir büyük asal sayı olmak üzere $P \in E(\mathbb{F}_p)$ noktasının mertebesi l olsun. E eliptik eğrisinin l ye göre gömme derecesi k olmak üzere $\mathbb{F}_{p^k}$ sonlu cismi üzerinde ayrık logaritma probleminin nasıl çözüldüğünün bilindiğini varsayalım ve $Q \in E(\mathbb{F}_p)$, P noktasının bir katı olsun. Bu durumda aşağıda verilen Menezes, Okamoto ve Vanstone (1993) tarafından verilen MOV algoritması P ve Q noktaları için eliptik eğri ayrık logaritma problemini çözer. Burada $E(\mathbb{F}_p)$ grubunda l mertebeli bir P noktası olduğundan $N = \#E(\mathbb{F}_{p^k})$ denirse $l \mid N$ dir.

3.11.3. MOV Algoritması

Input: $l > \sqrt{p} + 1$ özelliğinde bir asal sayı, mertebesi l olan bir $P \in E(\mathbb{F}_p)$ noktası, E eliptik eğrisinin l ye göre gömme derecesi k, P noktasının bir katı olan $Q \in E(\mathbb{F}_p)$ noktası.

1) $N = \#E(\mathbb{F}_{p^k})$ noktalarının sayısını hesapla.

2) $T \notin E(\mathbb{F}_p)$ olacak şekilde rastgele bir $T \in E(\mathbb{F}_{p^k})$ noktası seç.

3) $T' = (N/l)T$ değerini hesapla.

4) Eğer $T' = \boldsymbol{O}$ ise 2. adıma git.

5) T', l mertebeli bir nokta ise 6. adıma git.

6) $\alpha = e_l(P, T') \in \mathbb{F}_{p^k}^*$ ve $\beta = e_l(Q, T') \in \mathbb{F}_{p^k}^*$ Weil eşleştirmelerini hesapla. Eğer $\alpha = 1$ ise 2. adıma dön.

7) $\mathbb{F}_{p^k}^*$ grubunda α ve β için ALPni çöz, yani $\beta = \alpha^n$ olacak şekilde bir n kuvveti bul.

8) O halde $Q = nP$ dir ve böylece EEALP çözülmüş olur.

3.11.4. Uyarı 1. Yukarıda verilen MOV algoritması EEALPni çözer. Gerçektende, algoritmada kullanılan T' noktası P noktasından bağımsız olduğundan $\{T', P\}$ kümesi rankı 2 olan serbest $\mathbb{Z}/l\mathbb{Z}$-modül $E[l]$ için bir bazdır. Hatırlanacağı gibi Weil eşleştirmesi dejenere değildir, dolayısıyla $e_l(P, T')$ Weil eşleştirmesi $\mathbb{F}_{p^k}^*$ da bir birimin (aşikardan farklı) l. köküdür. Bir başka değişle

$$e_l(P, T')^r = 1 \text{ olması için gerek ve yeter koşul } l \mid r$$

olmasıdır. Eğer $Q = jP$ olduğu varsayılır ve j sayısının l modülüne göre değeri bulunmak istenirse MOV algoritması

$$e_l(Q, T') = e_l(P, T')^n$$

eşitliğini gerçekleyen bir n tamsayısını bulur. Weil eşleştirmesinin lineer olduğu kullanılarak

$$e_l(P, T')^n = e_l(Q, T') = e_l(jP, T') = e_l(P, T')^j$$

eşitlikleri elde edilir ve böylece $e_l(P, T')^{n-j} = 1$ dir. Bu ise P ve Q noktaları için $n \equiv j \pmod{l}$ değerinin EEALPni çözdüğünü gösterir.

2. MOV algoritmasının kullanışlı olup olmadığı k sayısına bağlıdır. Eğer k sayısı yeterince büyük, yani $k > (\ln p)^2$ ise MOV algoritmasını çözmek kolay değildir, yani bu algoritma kullanışlı değildir. Örneğin $p \approx 2^{160}$ olarak alınırsa $k > 4000$ olmak üzere ALPni $\mathbb{F}_{p^k}$ de çözmek gerekir. $\mathbb{F}_p$ cismi üzerinde tanımlı rastgele seçilen herhangi bir E eliptik eğrisinin gömme derecesi $(\ln p)^2$ den daha büyük olabileceğinden MOV algoritması kullanışlı değildir. Ancak bu algoritmanın kullanışlı olduğu, yani gömme derecesi küçük olan eliptik eğriler vardır. Aşağıdaki tanımda bu eğriler isimlendirilmiştir.

3.11.5. Tanım. E, $\mathbb{F}_p$ cismi üzerinde tanımlı bir eliptik eğri olmak üzere $E[p] = \{\boldsymbol{O}\}$ ise E eliptik eğrisine bir *süpersingüler eliptik eğri* denir.

3.11.6. Uyarı. $p \geq 5$ bir asal sayı olmak üzere $\mathbb{F}_p$ cismi üzerinde tanımlı E eliptik eğrisinin süpersingüler olması için gerek ve yeter koşul $\#E(\mathbb{F}_p) = p + 1$ olmasıdır. Bundan başka süpersingüler eğrilerin gömme dereceleri genellikle $k = 2$ veya $k \leq 6$ dır. Örneğin $E : y^2 = x^3 + x$ eğrisi, herhangi $p \equiv 3 \pmod 4$ asal sayısı için süpersingülerdir ve üstelik bu eğrinin gömme derecesi herhangi $l > \sqrt{p} + 1$ sayısı için 2'dir. Bu ise $E(\mathbb{F}_p)$ deki EEALPni çözmenin $\mathbb{F}_{p^2}^*$ daki ALPni çözmekten daha zor olmadığını gösterir.

4. BULGULAR VE TARTIŞMA

Bu bölümde eliptik eğrilerle eşleşen diziler ve eliptik eğri ayrık logaritma problemi arasındaki ilişkiler ortaya konularak eliptik eğri ayrık logaritma problemini daha kolay bir ayrık logaritma problemine dönüştüren iki algoritma verilecektir.

4.1. Eliptik Eğrilerle Eşleşen Diziler ve Eliptik Eğri Ayrık Logaritma Problemi

3. Bölümde görüldüğü gibi, E, karakteristiği 2'den farklı bir $\mathbb{F}$ cismi üzerinde tanımlı bir eliptik eğri ve $P \in E(\mathbb{F})$ olmak üzere $n \geq 1$ tamsayısı için P noktasının n katı nP,

$$nP = \left(\frac{G_n(P)}{F_n(P)^2}, \frac{H_n(P)}{F_n(P)^3} \right)$$

olarak yazılabilir. Silverman (Silverman, 2005), $\mathbb{F}_q$ sonlu cismi üzerinde tanımlı E eliptik eğrisinin bölüm polinomlarının $P \in E(\mathbb{F}_q)$ noktasında hesaplanan değerlerinin $(F_n(P))_{n\geq 0}$ dizisinin periyodik olduğunu göstermiştir. Shipsey ve Swart (2008), $(F_n(P))_{n\geq 0}$ dizisinin periyodiklik özelliklerini kullanarak EEALPni çözmek için alternatif bir algoritma vermişlerdir. Gezer ve Bizim (2019), $\mathbb{F}_q$ sonlu cismi üzerinde tanımlı E eliptik eğrisinin $P \in E(\mathbb{F}_q)$ noktasından elde edilen bölüm polinomlarının değerlerinin $(G_n(P))_{n\geq 0}$ ve $(H_n(P))_{n\geq 0}$ dizilerinin periyodik olduğunu göstermişlerdir. Bu kısımda $(G_n(P))_{n\geq 0}$ ve $(H_n(P))_{n\geq 0}$ dizilerinin periyodiklik özelliklerini kullanarak EEALPni çözmek için Shipsey ve Swart'ın algoritmasına benzer algoritmalar verilecektir.

Silverman (2005), $(F_n(P))_{n\geq 0}$ dizisinin periyodik olduğunu ve bu dizinin aşağıdaki teoremde verilecek olan simetri özelliklerini gerçeklediğini ispatlamıştır.

4.1.1. Teorem. E, $\mathbb{F}_q$ sonlu cismi üzerinde tanımlı bir eliptik eğri, $P \in E(\mathbb{F}_q)$ $N \geq 3$ mertebeli bir nokta olmak üzere $(F_n(P))_{n\geq 0}$, E eliptik eğrisinin bölüm polinomlarının P

noktasında hesaplanan değerlerinin dizisi olsun. Bu durumda, P noktasına bağlı olarak $a^N = b^2$ ve her $k, t \geq 0$ için

$$F_{kN+t}(P) = a^{kt} b^{k^2} F_t(P) \tag{4. 1}$$

olacak biçimde $a, b \in \mathbb{F}_q^*$ birimleri vardır (Silverman 2005).

Gezer ve Bizim (2019), $(G_n(P))_{n\geq 0}$ ve $(H_n(P))_{n\geq 0}$ dizilerinin simetri özelliklerini gerçeklediğini göstermişlerdir.

4.1.2.Teorem. E, $\mathbb{F}_q$ sonlu cismi üzerinde tanımlı bir eliptik eğri, $P \in E(\mathbb{F}_q)$ $N \geq 3$ mertebeli bir nokta olmak üzere $(G_n(P))_{n\geq 0}$ ve $(H_n(P))_{n\geq 0}$, E eliptik eğrisinin bölüm polinomlarının P noktasında hesaplanan değerlerinin dizileri olsun. Bu durumda, P noktasına bağlı olarak $a^N = b^2$ ve her $k, t \geq 0$ için

$$G_{kN+t}(P) = a^{2kt} b^{2k^2} G_t(P) \tag{4. 2}$$

$$H_{kN+t}(P) = a^{3kt} b^{3k^2} H_t(P) \tag{4. 3}$$

olacak biçimde $a, b \in \mathbb{F}_q^*$ birimleri vardır (Gezer ve Bizim 2019).

4.1.3. Algoritmalar

Shipsey ve Swart (2008), $|E(\mathbb{F}_q)| = q - 1$ olduğunda Teorem 4.1.1'i kullanarak EEALPni çözmek için alternatif bir algoritma vermişlerdir. Bu kısımda $(G_n(P))_{n\geq 0}$ ve $(H_n(P))_{n\geq 0}$ dizilerinin simetri özellikleri kullanılarak $E(\mathbb{F}_q)$ grubundaki bir EEALPni $\mathbb{F}_q^*$ grubundaki kolay bir ALPne indirgeyen iki algoritma verilecektir.

E, $\mathbb{F}_q$ sonlu cismi üzerinde tanımlı bir eliptik eğri, $P, Q \in E(\mathbb{F}_q)$ olmak üzere $Q = mP$ olsun. $P \in E(\mathbb{F}_q)$ noktasının mertebesi $N \geq 3$, $E(\mathbb{F}_q)$ grubunun mertebesi $q - 1$ olmak

üzere N sayısı $q - 1$ sayısının bir büyük asal çarpanı olsun. Bu durumda l bir küçük tamsayı olmak üzere $q - 1 = lN$ biçiminde yazılabilir. O halde Teorem 4.1.2. gereği,

$$G_{mq}(P) = G_{m + mlN}(P) = a^{2lm^2} b^{2l^2m^2} G_m(P),$$

$$G_{(m+1)q}(P) = G_{(m+1)+(m+1)lN}(P) = a^{2l(m+1)^2} b^{2l^2(m+1)^2} G_{m+1}(P)$$

dir. Yukarıdaki eşitlikler yeniden düzenlenirse

$$\frac{G_{(m+1)q}(P)G_m(P)}{G_{mq}(P)G_{m+1}(P)} = a^{2l(2m+1)} b^{2l^2(2m+1)^2}$$

$$= (a^{2l} b^{2l^2})^{2m+1} \tag{4. 4}$$

olduğu elde edilir. (3.2) ve (4. 2) gereği, $(a^{2l} b^{2l^2}) = G_q(P)/G_1(P) = G_q(P)/x(P)$ olduğundan (4.4) eşitliği

$$\left(\frac{G_q(P)}{x(P)}\right)^{2m+1} = \frac{G_{(m+1)q}(P)G_m(P)}{G_{mq}(P)G_{m+1}(P)} \tag{4. 5}$$

biçiminde yeniden yazılabilir. Dikkat edilirse m sayısı bilinmediğinden eşitliğin sağ tarafındaki herhangi bir değer hesaplanamaz. Diğer yandan (3.1) eşitliğinden

$$G_n(P) = x(nP)F_n(P)^2$$

olduğu elde edilir. Dolayısıyla

$$G_m(P) = x(mP)\, F_m(P)^2 = x(Q)\, F_m(P)^2, \tag{4.6}$$

$$G_{m+1}(P) = x((m+1)P)\, F_{m+1}(P)^2 = x(Q+P)\, F_{m+1}(P)^2, \tag{4.7}$$

$$G_{mq}(P) = x(mqP)\, F_{mq}(P)^2 = x(qQ)F_{mq}(P)^2, \tag{4.8}$$

ve $mP = Q$ ve $(m+1)P = P + Q$ olduğundan

$$G_{(m+1)q}(P) = x((m+1)qP)\, F_{(m+1)q}(P)^2 = x(q(Q+P))F_{(m+1)q}(P)^2 \tag{4.9}$$

dır. Diğer yandan Lemma 3.3.2 gereği,

$$F_{mq}(P) = F_m(P)^{q^2} F_q(mP) = F_m(P)^{q^2} F_q(Q) \tag{4.10}$$

dir ve $mP = Q$ ve $(m+1)P = P + Q$ olduğundan

$$F_{(m+1)q}(P) = F_{m+1}(P)^{q^2}\ F_q((m+1)P) = F_{m+1}(P)^{q^2} F_q(Q+P) \tag{4.11}$$

olduğu elde edilir. Böylece (4. 10) ve (4. 11) eşitlikleri kullanılarak (4. 8) ve (4. 9) teki eşitlikler

$$G_{mq}(P) = x(q(Q)F_m(P)^{2q^2} F_q(Q)^2 \tag{4. 12}$$

ve

$$G_{(m+1)q}(P) = x(q(Q+P)F_{m+1}(P)^{2q^2} F_q(Q+P)^2 \tag{4. 13}$$

olarak yeniden yazılabilir.

Şimdi (4. 11), (4. 12) eşitlikleri ve daha sonra (4. 6) ve (4. 7) eşitlikleri (4. 5) eşitliğinde yazılırsa

$$\left(\frac{G_q(P)}{x(P)}\right)^{2m+1} = \left(\frac{x(q(Q+P))F_{m+1}(P)^{2q^2} F_q(Q+P)^2 G_m(P)}{x(qQ)F_m(P)^{2q^2} F_q(Q)^2 G_{m+1}(P)}\right)$$

$$= \left(\frac{F_{m+1}(P)}{F_m(P)}\right)^{2(q^2-1)} \left(\frac{x(q(Q+P))x(Q)}{x(qQ)x(Q+P)}\right) \left(\frac{F_q(Q+P)}{F_q(Q)}\right)^2$$

eşitliği elde edilir. Dolayısıyla $\mathbb{F}_q^*$ grubunun mertebesi $q - 1$ olduğundan

$$\left(\frac{G_q(P)}{x(P)}\right)^{2m+1} = \left(\frac{x(q(Q+P))x(Q)}{x(qQ)x(Q+P)}\right) \left(\frac{F_q(Q+P)}{F_q(Q)}\right)^2$$

olarak bulunur. Böylece $G_q(P) = x(qP)\, F_q(P)^2$ olduğundan son eşitlik

$$\left(\left(\frac{G_q(P)}{x(P)}\right)^2\right)^m = \left(\frac{x(q(Q+P))x(Q)}{x(qQ)x(Q+P)}\right) \left(\frac{F_q(Q+P)}{F_q(Q)}\right)^2 \left(\frac{x(P)}{G_q(P)}\right)$$

$$= \left(\frac{x(q(Q+P))x(Q)x(P)}{x(qQ)x(qP)x(Q+P)}\right) \left(\frac{F_q(Q+P)}{F_q(Q)F_q(P)}\right)^2$$

olarak yeniden yazılabilir. Dikkat edilirse bu eşitliğin sağ tarafı hesaplanabilir. Böylece indeks hesabı yöntemi ile çözülebilecek bir $\alpha^m = \beta$ ayrık logaritma problemi elde edilir.

Benzer biçimde $(H_n(P))_{n\geq 0}$ dizisinin periyodik özelliklerinden bir $\mathbb{F}_q^*$ ALP eşitliği elde edilebilir. Böylece Teorem 4.1.2 kullanılarak ve yukarıdakine benzer biçimde hareket edilerek

$$\left(\left(\frac{H_q(P)}{y(P)}\right)^2\right)^m = \left(\frac{y(q(Q+P))\,y(P)y(Q)}{y(qQ)y(qP)y(P+Q)}\right)\left(\frac{F_q(P+Q)}{F_q(P)F_q(Q)}\right)^3$$

$\mathbb{F}_q^*$ ALP eşitliği elde edilir. Böylece başka bir $\gamma^m = \delta$ ALP elde edilir.

4.1.4. Uyarı. $G(1)(P) = x(P)$ olduğundan Teorem 4.1.2 gereği,

$$\frac{G_q(P)}{x(P)} = \frac{G_{lN+1}(P)}{x(P)} = \frac{a^{2l}\,b^{2l^2}x(P)}{x(P)} = a^{2l}\,b^{2l^2}$$

olarak bulunur. Şimdi $a^N = b^2$ olduğundan Teorem 4.1.2 gereği,

$$\frac{G_q(P)}{x(P)} = a^{2l}\,(b^2)^{l^2} = a^{2l}\,a^{Nl^2} = a^{2l}(a^{Nl})^l = a^{2l}(a^{q-1})^l = a^{2l}$$

dir. Diğer yandan, $\mathbb{F}_q^*$ da $a^{2lN} = a^{2(q-1)} = 1$ olduğundan a^2 nin mertebesi N sayısını böler. N bir asal sayı olduğu için $\frac{G_q(P)}{x(P)}$ nin mertebesi 1 ya da N olabilir. Eğer mertebe 1 ise saldırı başarısız olur. Ancak Shipsey ve Swart'ta olduğu gibi $\frac{G_q(P)}{x(P)} = 1$ olma olasılığı $\frac{1}{N}$ dir. Benzer biçimde $\frac{H_q(P)}{y(P)} = 1$ olma olasılığı da $\frac{1}{N}$ dır.

Şimdi aşağıdaki konjektürü verebiliriz.

4. 1. 5. Konjektür. E, $\mathbb{F}_q$ sonlu cismi üzerinde tanımlı bir eliptik eğri ve $P \in E(\mathbb{F}_q)$ noktasının mertebesi N olmak üzere $E(\mathbb{F}_q)$ grubunun mertebesi $q - 1$ ise

$$\frac{G_q(P)}{x(P)} = 1 \quad (\text{ya da } \frac{H_q(P)}{y(P)} = 1)$$

olma olasılığı $\frac{1}{N}$ dir (Gezer ve Turp 2019).

5. SONUÇ

Bu çalışmada ayrık logaritma problemi ve bu problemin çözümleri ele alınmıştır. Ayrık logaritma problemi birçok kriptografik yapıda kullanılmaktadır. Bu problemin çözümünün zorluk derecesi üzerinde çalışılan grubu bağlıdır. Örneğin, ayrık logaritma probleminin ($\mathbb{F}_p$, +) grubundaki çözümü oldukça kolaydır. Bununla birlikte ALPnin çözümü ($\mathbb{F}_p^*$, ·) grubu için zordur. Buna göre $O(p)$ mertebeli bir grupta ayrık logaritma problemi $O(\sqrt{p})$ adımda çözülebilir. $\mathbb{F}_p^*$ grubunda bu problemi çözmek için bilinen en iyi algoritma indeks hesabı yöntemidir ve bu yöntem ALPni c belli bir sabit olmak üzere

$$\exp(c\sqrt[3]{(\log p)(\log\log p)^2})$$

zamanda çözer. Bu zaman altüstel zaman olarak bilinmektedir. Genel olarak ALPnin üstel bir zamanda çözülebileceği bir G grubu alınır. Bu çalışmada ayrık logaritma problemi özellikle $\mathbb{F}_p^*$ sonlu grubu ve sonlu bir cisim üzerinde tanımlı bir eliptik eğrinin üzerindeki noktaların oluşturduğu $E(\mathbb{F}_p)$ grubu için ele alınmıştır.

Literatürde EEALPni daha kolay bir ALPne dönüştüren algoritmalar verilmektedir. Çalışmada eliptik eğriler ile eşleşen diziler ve eliptik eğri ayrık logaritma problemi arasındaki ilişkiler ortaya konulmuş ve EEALPni daha kolay bir ALPne dönüştüren iki algoritma verilmiştir. Daha sonra konu ile ilgili bir konjektür verilmiştir.

KAYNAKLAR

Diffie, W., Hellman, M.E. 1976. New directions in cryptography, *IEEE Trans. Information Theory*, 22(6): 644-654.

Elgamal, T. 1985. A public key cryptosystem and a signature scheme based on the discrete logarithms. *IEEE Trans. Information Theory*, 31(4): 469-472.

Fraleigh, J. B. 2003. A first course in abstract algebra. Addison-Wesley, 513 pp.

Gezer, B., Bizim, O. 2019. Sequences generated by elliptic curves. *Acta Arithmetica*, 188(3): 253-268.

Gezer, B., Turp, S. 2019. The elliptic curve discrete logarithm problem and sequences associated to elliptic curves. *Journal of Mathematical Cryptology* adlı dergiye sunuldu.

Hoffstein, J., Piper, J., Silverman, J. H. 2008. An introduction to mathematical cryptography. Springer, New York, USA, 523 pp.

Koblitz, N. 1987. Elliptic curve cryptosystems. *Math. Comp*., 48(177): 203-209.

Mazur, B., Tate, J. 1991. The *p*-adic sigma function. *Duke Math. J*., 62: 663-688.

Menezes, A. J., Okamoto, T., Vanstone, S. A. 1993. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Trans. Inform. Theory*, 39(5):1639-1646.

Miller, V. S., 1986. Use of elliptic curves in cryptography: In advances in cryptology-Crypto'85 (Santa Barbara, CA), volume 218 of lecture notes in comput. sci., Ed., Williams, H. C., Berlin, Germany, pp: 417-426.

Rivest, R. L., Shamir, A., Adleman. L. 1978. A method for obtaining digital signatures and public-key cryptosystems. *Comm. ACM*, 21(2):120-126.

Schoof, R., 1985. Elliptic curves over finite fields and the computation of square roots mod *p*. *Math. Comp*. 44(170): 483-494.

Shipsey, R., Swart, C. 2008. Elliptic divisibility sequences and elliptic curve discrete logarithm problem, Cryptology ePrint Archieve. https://eprint.iacr.org/2008/444.pdf-(Erişim tarihi: 12.04.2019).

Shoup, V. 2001. OAEP reconsidered. In advances in cryptology-Crypto 2001 (Santa Barbara, CA), volume 2139 of lecture notes in comput. sci., Ed., Williams, H. C., Berlin, Germany, pp: 239-259.

Silverman, J. H. 2005. *p*-adic properties of division polynomials and elliptic divisibility sequences, *Math. Ann*., 332: 443-471, addendum 473-474.

Silverman, J. H. 2009. The arithmetic of elliptic curves. Springer, New York, USA, 513 pp.

Silverman, J. H., Tate J. 1994. Rational points on elliptic curves, Springer, New York, USA, 267 pp.

Washington, L. C. 2003. Elliptic curves: Number theory and cryptography. Chapman & Hall / CRC, Boca Raton, FL, 428 pp.

Western, A. E., Miller, J. C. P. 1968. Tables of indices and primitive Roots: Royal Society Mathematical Tables, Vol. 9. Cambridge University Press, London, UK, 440 pp.

ÖZGEÇMİŞ

Adı Soyadı	: Semiha TURP
Doğum Yeri ve Tarihi	: BURSA – 10.08.1993
Yabancı Dili	: İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise	: İMKB Gürsu Anadolu Lisesi -2011
Lisans	: Uludağ Üniversitesi -2016
Yüksek Lisans	: Uludağ Üniversitesi Fen Bilimleri Enstitüsü-2019

Çalıştığı Kurum/Kurumlar ve Yıl	:
İletişim (e-posta)	: semihaturp@outlook.com
Yayınları	: The elliptic curve discrete logarithm problem and sequences associated to elliptic curves. *Journal of Mathematical Cryptology* adlı dergiye sunuldu.