KRİPTOLOJİ BİLİMİ VE ANAHTAR DAĞITIM ŞEMALARI

Ferhat POLAT

Yüksek Lisans Tezi

Matematik Ana Bilim Dalı

Dr. Öğr. Üyesi Abdullah ÇAĞMAN

AĞRI-2022

T.C.
AĞRI İBRAHİM ÇEÇEN ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
MATEMATİK ANA BİLİM DALI

Ferhat POLAT

KRİPTOLOJİ BİLİMİ VE ANAHTAR DAĞITIM ŞEMALARI

YÜKSEK LİSANS TEZİ

TEZ YÖNETİCİSİ
Dr. Öğr. Üyesi Abdullah ÇAĞMAN

AĞRI-2022

TEZ KABUL VE ONAY TUTANAĞI
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ MÜDÜRLÜĞÜNE

Dr. Öğr. Üyesi Abdullah ÇAĞMAN danışmanlığında, Ferhat POLAT tarafından hazırlanan bu çalışma 28/07/2022 tarihinde aşağıdaki jüri tarafından Matematik Ana Bilim Dalı'nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Başkan : Dr. Öğr. Üyesi Sait TAŞ İmza: ……..
Jüri Üyesi : Dr. Öğr. Üyesi Abdullah ÇAĞMAN İmza: ……..
Jüri Üyesi : Dr. Öğr. Üyesi Kadirhan POLAT İmza: ……..

Yukarıdaki imzalar adı geçen öğretim üyelerine ait olup;

Enstitü Yönetim Kurulunun …/…/2022 tarih ve . . . . / . . . . . nolu kararı ile onaylanmıştır.

…. /……/…….
Prof. Dr. İbrahim HAN
Enstitü Müdürü

ÖZET

YÜKSEK LİSANS TEZİ

KRİPTOLOJİ BİLİMİ VE ANAHTAR DAĞITIM ŞEMALARI

Tez Danışmanı: Dr. Öğr. Üyesi Abdullah ÇAĞMAN

Jüri: Dr. Öğr. Üyesi Sait TAŞ
Dr. Öğr. Üyesi Abdullah ÇAĞMAN
Dr. Öğr. Üyesi Kadirhan POLAT

Haberleşme ve iletişim; geçmişten günümüze kadar süregelen olguları içermektedir. Haberleşmede insanlar, karşı tarafa aktardığı bilgileri 3. kişilerle ya da art niyetli insanların erişememesi için çeşitli şifreleme işlemleri uygulamışlardır. Bilginin aktarımında kullanılan şifreleme işlemlerinin temelinde şüphesiz matematiksel ifadelerin önemi çok büyük. Bu konuyla alakalı daha önce yapılan bilimsel çalışmaların ışığında, konuya farklı bir açıdan yaklaşıldı. İlk bölümde şifreleme bilimi olan kriptoloji alanıyla ilgili genel bilgiler verilip, 2. bölümde matematiksel ifadelere kısaca değinilip, 3. bölümde de bilgi güvenliği unsurları ve kripto sistemlere değinildi. Son bölümde ise tezimizin ana temasını oluşturan anahtar dağıtım şemalarından ve protokollerinden bahsedildi.

2022, 57 sayfa

Anahtar sözcükler: Kriptoloji, şifreleme, deşifreleme, anahtar dağıtım şemaları ve protokolleri.

ABSTRACT
MASTER'S THESIS

SCIENCE OF CRYPTOLOGY AND KEY DISTRIBUTION SCHEMES

Advisor Of Thesis: Assist. Prof. Dr. Abdullah ÇAĞMAN

Jury: Assist. Prof. Dr. Sait TAŞ
Assist. Prof. Dr. Abdullah ÇAĞMAN
Assist. Prof. Dr. Kadirhan POLAT

Communication and telecommunucation; it includes facts from past to present. In communication, people have applied various encryption processes so that the information they transfer to the other party cannot be accessed by third parties or malicious people. Undoubtedly, the importance of mathematical expressions is very great on the basis of encryption processes used in the transfer of information. In the light of previous scientific studies on this subject, the subject was approached from a different angle. In the first part, general information about the field of cryptography, which is the science of encryption, was given, in the second part, mathematical expressions were briefly mentioned, and in the third part, information security elements and cryptosystems were mentioned. In the last part, key distribution schemes and protocols, which constitute the main theme of our thesis, were mentioned.

2022, 57 pages

Keywords: cryptology, encryption, decryption, key distribution scheme and protocols.

TEŞEKKÜR

Yüksek lisans eğitimim boyunca, benden bilgi ve deneyimlerini esirgemeyen, çalışmalarımın tamamlanabilmesi için her türlü imkânı sağlayan ve bana her zaman her türlü desteği sunan danışman hocam Sayın Dr. Öğr. Üyesi Abdullah ÇAĞMAN'a teşekkürlerimi sunarım.

Eğitimimin tüm süreçlerinde her türlü destekleriyle beni hiç yalnız bırakmayan anne-babama, kardeşlerime ve oğlum Yusuf Kayra'ya teşekkür ederim.

02/08/2022

Ferhat POLAT

İçindekiler

ÖZET ..... **ii**
ABSTRACT ..... **iii**
TEŞEKKÜR ..... **iv**
SİMGELER VE KISALTMALAR DİZİNİ ..... **vii**
ŞEKİL VE ÇİZELGELER DİZİNİ ..... **ix**
TABLOLAR LİSTESİ ..... **x**
1.GİRİŞ **1**
1.1. Şifreleme İle İlgili Temel Kavramlar 1
1.2. Kriptolojinin Tarihçesi 3
2. ŞİFRELEMEDE TEMEL MATEMATİKSEL KAVRAMLAR **5**
2.1. Fonksiyon Kavramı 5
2.2. Modüler Aritmetik 6
2.3. Grup Tanımı ve Özellikleri 8
2.4. Euler Fonksiyonu 9
3. ŞİFRELEME SİSTEMLERİ **10**
3.1. Bilgi Güvenliği Unsurları 10
3.2. Kriptosistemler 11
3.2.1. Açık anahtarlı (Asimetrik) sistemler 11
3.2.2. Gizli anahtarlı (simetrik) sistemler 12
3.2.3. Hibrit Sistemler 14
3.3. Klasik Kriptoloji 15
3.3.1 Kaydırma şifresi (The Shift Cipher) 16
3.3.3. Affine Şifreleme Sistemi (Affine Cipher) 20
3.3.4. Vigenere şifresi 22
3.3.5. Hill Şifrelemesi (Blok Şifreleme) 23
3.3.6 Permütasyon Şifresi 24
3.3.7 Akış Şifreleri 26
4. KRİPTOSİSTEMLER **28**
4.1. En Çok Kullanılan Gizli Anahtarlı (Simetrik) Kriptografi Algoritmaları 28
4.1.1. Veri Şifreleme Standardı (Data Encryption Standart (DES)) 30
4.1.2. Uluslararası Veri Şifreleme Algoritması (International Data Encryption Algorithm (IDEA)) 32
4.1.3. BlowFish 32

4.1.4. RC5 32
4.1.5. CAST-128 33
4.1.6. AES 33
4.2. En Çok Kullanılan Açık Anahtar (Asimetrik) Algoritmaları 34
4.2.1. Diffie-Hellman 35
4.2.2. RSA 37
4.2.3. El Gamal 39
4.3. Anahtar Dağıtım Şemaları 41
4.3.1 Needham-Schroeder Anahtar Dağıtım Şeması 41
4.3.2. Denning-Sacco Anahtar Dağıtım Şeması 43
4.3.3 Kerberos 44
4.3.4. Bellare-Rogaway Şeması 49
5. SONUÇ VE ÖNERİLER 53
KAYNAKLAR 55
ÖZGEÇMİŞ ..... Hata! Yer işareti tanımlanmamış.

SİMGELER VE KISALTMALAR DİZİNİ

$<$	Küçüktür
$>$	Büyüktür
$\leq$	Küçük veya Eşittir
$\geq$	Büyük veya Eşittir
$\notin$	Elemanı Değildir
$\equiv$	Denktir
$\neq$	Eşit Değildir
$\forall$	Her
φ	Phi
π	Pi
∞	Sonsuz
$\Longrightarrow$	İse
$\Leftrightarrow$	Ancak Ve Ancak
AES	İleri Şifreleme Standardı
DES	Veri Şifreleme Standardı
IDEA	Uluslararası Veri Şifreleme Algoritması
MAC	Mesaj Doğrulama Kodu
RSA	Rivest Shamir Aderman
DSA	Dijital İmza Algoritması
RCA	Kök Sertifika Otoritesi
MTI	Matsumoto Takashima Imai
$EBOB$	En Büyük Ortak Bölen
$EKOK$	En Küçük Ortak Kat
$\mathbb{N}$	Doğal Sayılar Kümesi

$\mathbb{R}$	Reel Sayılar Kümesi
f'	f Fonksiyonunun Birinci Mertebeden Türevi
f''	f Fonksiyonunun İkinci Mertebeden Türevi
max	Maksimum
min	Minimum

ŞEKİL VE ÇİZELGELER DİZİNİ

Şekil 1.1. Şifreleme Olayının Görselleştirilmesi 3
Şekil 3.1. Asimetrik Şifreleme 11
Şekil 3.2. Simetrik Şifreleme 13
Şekil 4.1. Klasik Feistel Network 29
Şekil 4.2. DES Algoritmasının Genel Yapısı 31
Şekil 4.3. DES Algoritmasının Genel İşleyişi 31
Şekil 4.4. Diffie-Hellman Anahtar Anlaşması 36
Şekil 4.5. Needham-Schroeder Bilgi Akışı 43
Şekil 4.6. Kerberos'un Bilgi Akışı 45

TABLOLAR LİSTESİ

Tablo 3.1. Kriptosistemlerin Güvenlik Unsuru Açısından Karşılaştırılması 14
Tablo 3.2. Türk Alfabesinin $\boldsymbol{mod\ 29}$'a Göre Tamsayı Karşılıkları 17
Tablo 3.3. Örnek π Fonksiyonu 19
Tablo 3.4. Örnek $\boldsymbol{\pi^{-1}}$ Fonksiyonu 19
Tablo 3.5. Örnek Vigenere Şifrelemesi 23
Tablo 3.6. Örnek Vigenere Deşifrelemesi 23

1.GİRİŞ

Haberleşme ve iletişim; tarihten günümüze kadar süregelen olguları içermektedir. Haberleşmede insanlar, karşı tarafa aktardığı bilgileri 3. kişilerle ya da art niyetli insanların ulaşmaması için çeşitli yol ve yöntemler denemişlerdir. Bu yöntemler (kodlama, şifreleme) tarihi serüven içerisinde, en ilkelinden başlayarak, günümüz en ileri teknolojileri sayesinde çok farklı bir boyuta ulaşmıştır. Hatta işin boyutu devletlerin ve ulusların en önemli politikaları haline gelmiştir. Karşılıklı taraflar arasındaki iletişimin sağlıklı ve güvenli yürütülmesi adına eskiden beri süregelen birçok faktör devreye girmiştir. Bu faktörler iletinin kodlanması veya şifrelenmesi ve bu vesileyle düşman kişinin eline geçmesini veya geçse bile iletinin içeriğini anlamasını zorlaştırmaktadır. Teknolojinin hızlı gelişimi, özellikle milenyum çağındaki aşırı seviyedeki ilerleyişi beraberinde de birçok olumsuz ögeleri de barındırmaktadır. Normal zamanda basit bir şifreleme ile karşı tarafa ileteceğimiz bilgiyi, şimdi daha kompleks daha güçlü şifreleme ile iletmek zorundayız. Her zorunluluk, insanları farklı alternatiflere yönelttiği gibi şifreleme alanında da bu zorunluluklardan dolayı kriptoloji bilimi şekillenmiş ve son olarak günümüzdeki haline (çeşitliliğine) bürünmüştür. Konuyla alakalı terimleri tanıyarak teze devam edilecektir.

1.1. Şifreleme İle İlgili Temel Kavramlar

Şifreleme (encryption): Karşı tarafa iletilecek olan mesajın içeriğinin çeşitli yöntem veya tekniklerle gizlenerek aktarılması durumudur.

Deşifreleme (decryption): Gönderici tarafından gönderilen şifreli mesajı alıcının çözümlemesi durumu.

Şifre kırma (crack): İçeriği gizlenmiş olan herhangi bir mesajın içeriğine ulaşmak için uğraşılan alandır.

Saldırgan (adversary): Mesajın alıcısı veya göndericisi olmadığı halde mesaja ulaşmaya çalışan kişidir (Külen, 2013).

Saldırı (atack): Kötü niyetli kişilerin, gönderilen şifreli mesajın içeriğini çözmek için yaptığı girişimdir.

Anahtar (key): Şifrelemede veya şifre çözmede kullanılan algoritmanın bilinmeyen kısmını oluşturan bölümüdür (Külen, 2013).

Açık anahtar (publickey): Asimetrik şifreleme sisteminin anahtar türüdür. İsteyen herkesin ulaşabileceği bir anahtar tipidir.

Gizli anahtar (private key): Yine aynı şekilde asimetrik şifrelemede kullanılan ancak sadece kullanıcısının ulaşabileceği anahtar tipidir.

Açık metin (plaintext): Şifrelenmemiş, kodlanmamış asıl metin.

Şifreli metin (ciphertext): Asıl (orjinal) metnin çeşitli yöntemlerle şifrelenmesi sonucu ortaya çıkan metindir.

Sunucu (server): Verileri saklayıp düzenleyip dijital imzaları karşı tarafa ileten yetki sahibidir (Kocabıyık, 2021).

İstemci (client): Sunucu tarafından hizmet alarak bilgi paylaşımı yapılan kimsedir (Yıldırım, 2006).

Gönderici (sender): Mesajı(iletiyi) oluşturan ve ulaştırmak istediği kişiye gönderen kişidir.

Alıcı (receiver): Gönderilen mesajın muhatabı.

Kanal (channel): Mesajın gönderildiği araçtır.

Şifreleme algoritması: Şifreleme ve deşifreleme işlemlerinde kullanılan akış diyagramı ve matematiksel işlemlerin bütünüdür (Kocabıyık, 2021).

Kriptografi: Kriptografi, köken olarak yunanca gizli saklı anlamına gelen ***kryptos*** ve yazmak anlamına gelen ***graphein*** kelimelerinden oluşturulmuştur. Kriptografi; gizlilik, kimlik denetimi, bütünlük gibi bilgi güvenliği kavramlarını sağlamak için çalışan matematiksel yöntemler bütünüdür. Bu yöntemler, bir iletiyi karşı tarafa iletimi esnasında karşılaşılabilecek saldırılardan iletiyi, göndericiyi ve alıcıyı koruma amacı taşır. Bir başka deyişle kriptografi, okunabilir durumdaki bir iletinin istenmeyen taraflarca okunamayacak bir hale dönüştürülmesinde kullanılan tekniklerin tamamıdır (Çimen, 2007).

Kriptanaliz: Kriptanaliz, nasıl çalıştıklarını anlamak ve onları yenmek veya zayıflatmak için teknikler bulmak ve geliştirmek amacıyla şifreli metin, şifreler ve şifreleme sistemlerinin incelenmesidir.

Kriptoloji: Hem kriptografi hem de kriptanaliz olayını beraber barındıran bir kavramdır. Yani hem şifreleme hem de şifre çözme (metni anlamlaştırma) olaylarının genel tanımıdır.

Şekil 1.1. Şifreleme Olayının Görselleştirilmesi

1.2. Kriptolojinin Tarihçesi

Kriptografinin tarih sahnesindeki bariz olarak ilk örnekleri Julius Caesar'ın MÖ 60-50 yıllarında Roma alfabesindeki harflerin yerlerini değiştirerek oluşturduğu şifreleme yöntemini, devlet haberleşmesinde kullanması örnek gösterilir. Bu metot, şifrelenecek olan metindeki her harfin alfabede kendisinden 3 sonraki harfle değiştirilmesine dayanmaktadır (Ersin, 2006).

Kriptografinin daha sonrasındaki gelişimi ise şöyle devam ettiği bilinir (Ersin 2006).

- 1586, Blaise de Vigenere (1523-1596) şifreleme hakkında bir kitap kaleme aldı. İlk defa bu kitapta açık metin ve şifreli metin için otomatik anahtarlama yönteminden bahsedildi (Weis, 2007).
- 1623'de Sir Francis Bacon, 5-bit ikili kodlamayla karakter tipi değişikliğine dayanan stenografı buldu (Aydoğan, 2014).
- 1920'li yıllarda Amerikan federal soruşturma bürosu(FBI), kaçakçıların iletişimini çözmek için bir büro kurdu.
- 2. Dünya Savaşı yıllarında Almanların gizli haberleşmede kullandığı enigma makinesi, İngiliz Alan Turing ve ekibi tarafından çözüldü ve bu durum, savaşın seyrinin değişmesine sebep oldu.
- 1970'lerde Horst Feistel, Data Encryption Standard'ın (DES) temelini oluşturan Lucifer algoritmasını geliştirdi (Koblitz, 2006).
- 1976 yılında, kriptoloji bilimine önemli katkılar sağlayan Whitfield Diffie ve Martin Hellman, açık anahtar sistemini konu alan çalışmalarını yayınladılar.
- 1978'de Ronald L. Rivest, Adi Shamir ve Leonard M. Adleman, Rivest-Shamir-Adleman (RSA) algoritmasını buldular (Bahçetepe, 2006).

- 1990’da Xuejia Lai ve James Massey, açılımı International Data Encryption Algorithm olan IDEA algoritmasını buldular.
- 1991’de phil zimmerman , açılımı Pretty Good Privacy olan PGP sistemini yayınladı (Bahçetepe, 2006).
- 1997’de ABD’nin NIST kurumu DES’inyerini alacak bir simetrik algoritma için yarışma açtı (Bahçetepe, 2006).
- 2001’de NIST’in yarışmasını kazanan Belçikalı Joan Daemen ve Vincent Rijmen’e ait Rijndael algoritması, Advanced Encryption Standard (AES) adıyla standart haline getirildi (Ersin 2006; Başkök 2007).

2. ŞİFRELEMEDE TEMEL MATEMATİKSEL KAVRAMLAR

Bu bölümde kullanılacak olan matematiksel tanım, teorem ve kavramlar ile alakalı verilerde Balcı (1985), Altındiş (1999), Şenay (2007), Gilbert (2009) ve Çallıalp (2009)'dan faydalanılmıştır.

2.1. Fonksiyon Kavramı

A ve B iki küme olsun. A'dan B'ye olan bir f bağıntısı aşağıdaki koşulları yerine getiriyorsa f'ye A'dan B'ye bir fonksiyon denir.

- $\forall x \in A$ için $(x, y) \in f$ olmak üzere B' de en az bir y elemanı olur.
- $(x, y) \in f$ ve $(x, z) \in f$ ise $y = z$'dir.

f, A'dan B'ye fonksiyonu $f: A \to B$ şeklinde gösterilir. Bu gösterimde A'ya tanım kümesi, B'ye ise değer kümesi denir.

$f: A \to B$ fonksiyonunda $(x, y) \in f$ ise bu ikili $y = f(x)$ şeklinde de gösterilebilir.

Tanım 2.1.1: $f: A \to B$ bir fonksiyon ise A'nın f altındaki görüntü kümesi;

$f(A) = \{f(x): x \in A\}$ şeklinde tanımlanır.

Tanım 2.1.2: $f: A \to B$fonksiyonu için, $f(A) = B$ ise f'ye örten fonksiyon denir. Buna göre f örtense , $\forall y \in B$ için, $f(x) = y$ olacak şekilde en az bir $x \in A$ vardır.

Tanım 2.1.3: $f: A \to B$ bir fonksiyon ve $x_1 , x_2 \in A$ olmak üzere;

$x_1 \neq x_2$ için $f(x_1) = f(x_2)$ ise f'ye birebir fonksiyon denir.

Tanım 2.1.4: f'nin birebir ve örten olması durumunda f^{-1} dönüşümüne f dönüşümünün tersi denir. Yani $f: A \to B$ birebir ve örtense $f^{-1} : B \to A$ dönüşümü $x \in A$ ve $y \in B$ için ;

$x = f^{-1}(y) \Leftrightarrow y = f(x)$ olarak tanımlanır.

2.2. Modüler Aritmetik

Asimetrik (açık anahtarlı) şifrelerin oluşumunda ve deşifre edilmesinde modüler aritmetiğin rolü oldukça büyüktür. Öncelikle modüler aritmetiğin tanımıyla başlanmıştır.

Tanım 2.2.1: $a, b \in \mathbb{Z}$ ve $n \neq 0$ bir tamsayı olmak üzere, $a - b$ 'nin farkı n'nin bir k katı oluyorsa bu durumda;

$$a - b = k.n \qquad (1)$$

ya da

$$a \equiv b(mod\ n) \qquad (2) \quad \text{olur.}$$

Modüler Aritmetikle İlgili Bazı Özellikler

1.Toplama

$$(\ a\ mod\ n\ ) + (\ b\ mod\ n\ ) = (\ a + b\ )\ mod\ n$$

2. Çıkarma

$$(\ a\ mod\ n\ ) - (\ b\ mod\ n\ ) = (\ a - b\ )\ mod\ n$$

3. Çarpma

$$(\ a\ mod\ n\ )\ x\ (\ b\ mod\ n\ ) = (\ a\ x\ b\ )\ mod\ n$$

4. Bölme

$\frac{(\ a\ mod\ n\ )}{(\ b\ mod\ n\ )} = a\ x\ b^{-1} \quad mod\ n$ (b'nin tersi ile çarpma işlemi yapılır.)

5. Birleşme

$[\,(a+b)+c\,]\ mod\ n = [\,a+(b+c)\,]\ mod\ n$

6. Değişme

$(a+b)\,mod\ n = (b+a)\ mod\ n$

7. Geçişlilik

$a \equiv b\ mod\ n$ ve $b \equiv c\ mod\ n$ ise $a \equiv c\ mod\ n$

8. Dağılma

$[\,a\ x\ (b+c)\,]\ mod\ n = [\,(a\ x\ b)+(a\ x\ c)\,]\ mod\ n$

9. Etkisiz eleman

- $(0+a)\ mod\ n = a\ mod\ n$ (toplama işlemi için)
- $(1\ x\ a)\ mod\ n = a\ mod\ n$ (çarpma işlemi için)

10. Toplam invers ($-a$)

∀a ∈ $\mathbb{Z}_n$ için; öyle bir b vardır ki; $a+b=0\ mod\ n$'dir.

Teorem : n pozitif tamsayı ve $a, b, c \in \mathbb{Z}$ olmak üzere aşağıdaki önermeler doğrudur.

1. $a \equiv a\ mod\ n$ (**yansıma özelliği**)
2. $a \equiv b\ mod\ n \Rightarrow b \equiv a\ mod\ n$ (**simetri özelliği**)

3. $a \equiv b\ mod\ n,\ b \equiv c\ mod\ n \Rightarrow a \equiv c\ mod\ n$ (**geçişme özelliği**)

2.3. Grup Tanımı ve Özellikleri

G boş olmayan bir küme ve $(*)$ da G üzerinde tanımlı bir ikili işlem olsun. Eğer aşağıdaki şartlar sağlanıyorsa $(G,*)$ cebirsel yapısına grup denir.

1. Her $a,b \in G$ için $a * b \in G$'dir (kapalılık özelliği).

2. Her $a,b,c \in G$ için $(a * b) * c = a * (b * c)$ (birleşme özelliği).

3. Her $a \in G$ için $a * e = e * a = a$ olacak şekilde bir $e \in G$ vardır (birim eleman özelliği).

4. Her $a \in G$ için $a * b = b * a = e$ olacak şekilde bir $b \in G$ vardır (ters eleman özelliği).

NOT: Bu şıktaki b elemanına aynı zamanda a'nın tersi denir ve $b = a^{-1}$ şeklinde de gösterilir.

5. Her $a,b \in G$ için $a * b = b * a$ ise (değişme özelliği) bu gruba abelyen (değişmeli) grup denir.

Tanım: Bir G grubu için, $G =< M >$ olacak şekilde bir $M \subset G$ alt kümesi bulunabiliyorsa, G'ye M ile üretilmiş grup denir. Eğer M sonlu bir küme ise G'ye sonlu üretilmiş grup ve $M = \{a\}$ tek elemanlı bir küme ise G'ye a ile üretilmiş devirli gr.up denir ve $G =< a >$ yazılır (Çallıalp, 1986).

Tanım: Bir G gubunun elemanlarının sayısına (eğer G sonsuz ise) G'nin mertebesi denir ve $|G|$ ile gösterilir. Eğer G sonsuz bir küme ise $|G| = \infty$ ile gösterilir.

2.4. Euler Fonksiyonu

Tanım 2.4.1: n bir pozitif tamsayı, $1 \leq a \leq n$ ve $(n, a) = 1$ olan a tamsayılarının sayısı $\phi(n)$ ile gösterilir ve Euler fonksiyonu olarak adlandırılır.

Euler fonksiyonunun özellikleri

1. Eğer p asal sayı ise $\phi(p) = p - 1 = p.(1 - \frac{1}{p})$

2. p asal sayı ve $a \in N$ ise ; $\phi(p^a) = p^a - p^{a-1} = p^a(1 - \frac{1}{p})$

3. $(m, n) = 1$ ise $\phi(mn) = \phi(m).\phi(n)$

3. ŞİFRELEME SİSTEMLERİ

3.1. Bilgi Güvenliği Unsurları

Haberleşmede bilgi güvenliğinin sağlanabilmesi için kullanıcılar ve sağlayıcılardan beklenenler vardır. Güvenliğin temel unsurları olan gizlilik, bütünlük, kimlik doğrulama, inkar edilemezlik ve haberleşmede süreklilik ile bu beklentiler karşılanabilir (İmamoğlu, 2011). Bilgi güvenliği unsurlarının sağlanabilmesi adına aşağıda verilen kavramlar büyük ölçüde önem arz etmektedir**.**

1. Gizlilik: Karşı tarafa aktarmak istediğimiz iletiyi çeşitli şifreleme algoritmaları kullanarak, istenmeyen kişilerin eline geçmesini önlemek veya geçse bile bu istenmeyen kişilerin iletiyi anlamlı hale (deşifreleme) getirmesini imkansız kılmaktadır. Şifrelemede en temel özelliklerden birisi olan gizlilik unsuru, gönderici ile alıcı arasındaki iletişimin sağlıklı bir şekilde yürütülmesini sağlar.

2. Bütünlük: Aktarılmak istenen iletinin içeriğinin orijinal halinin karşı tarafa ulaştırılmasıdır. Bu yüzden iletinin içeriğinin orijinal hali bozulmamış, değiştirilmemiş olması önemlidir.

3. Süreklilik: Sunucu ve alıcı arasındaki haberleşmenin kesintisiz olması ve bir an bile sekteye uğramaması durumudur. Taraflar arasındaki iletişimin, 3. Kişiler (kötü niyetli) tarafından tehlikeye düşürülmemesi adına gerekli önlem ve tedbirler alınmalıdır.

4. Kimlik doğrulama: Veri iletiminde sistemin gerçek kişileri yani, sunucu ile alıcıyı tanıyıp sisteme öyle kabul etmesidir. Sistemin güvenliği açısından bu son derece önemlidir. Günümüzde de kimlik doğrulama adına çok çeşitli varyasyonlar kullanılmaktadır.

5. İnkar edilememezlik: Taraflar arasında gerçekleşen mesajlaşmanın sistem tarafından kayıt altına alınıp, daha sonra taraflardan birinin veya her ikisinin de itirazlarına karşılık ortaya konan belgelerdir. Diğerlerinde olduğu gibi bu güvenlik unsuru da son derece önemlidir. Özellikle bankacılık ve finansman anlamında tarafların mağdur olmaması adına önemli bir özelliktir.

6. İzlenebilirlik: Sunucu ile alıcı arasındaki haberleşmenin sonraki zamanlardaki ihtiyacı halinde kullanılması adına kayıt altına alınması durumudur.

3.2. Kriptosistemler

Kriptosistemler genel olarak; gizli anahtarlı (simetrik) sistemler ve açık anahtarlı (asimetrik) sistemler olmak üzere temelde 2 ana başlık altında incelenir.

3.2.1. Açık anahtarlı (Asimetrik) sistemler

Gönderici ile alıcı arasındaki mesajlaşmada, iletinin şifrelenmesi ve deşifrelenmesinde kullanılan anahtar türleri birbirinden farklıdır. Bu anahtarlardan birine açık anahtar (public key), diğerine ise özel anahtar (private key) adı verilir. Bu iki anahtar türü birlikte imal edilir ve aralarında mantıksal olarak sıkı bir ilişki vardır (Alkan ve Sağıroğlu 2005).

Asimetrik anahtarlı sistemlerde her kullanıcının bir anahtar çifti vardır. Bu anahtarlardan biri kendi şahsi anahtarıdır ve kesinlikle muhafaza edilmesi gerekir. Diğeri ise açık anahtardır ve bu kişiye mesaj göndermek isteyen kişiler tarafından kullanılır. Şöyle özetleyecek olursak; gönderici, göndermek istediği iletiyi, alıcının açık anahtarıyla şifreler. Alıcı ise, gelen iletiyi kendi özel anahtarıyla açar.

Bu algoritma türüne örnek olarak; RSA, ECC, Diffie-Hellman ve El-Gamal verilebilir.

Şekil 3.1. Asimetrik Şifreleme (Frolov, 2021)

Asimetrik (açık anahtarlı) şifrelemenin avantajları

- İnkar edilememezlik, kimlik doğrulama ve gizlilik sağlar.
- Anahtar dağıtımı simetrik şifrelemeye göre daha kolaydır.
- Anahtarın çalınma ihtimaline binaen simetrik anahtara göre daha güvenlidir.
- Asimetrik şifrelemenin bir diğer önemli avantajlarından biri ise veri şifreleme ve deşifreleme işlemlerini yapmasının yanında dijital imza gerçekleştirilmede kullanılmasıdır. Alıcıya gelen şifreli metnin, doğru (asıl) kişiden gelip gelmediğini teyit etmek için bu dijital imzalardan faydalanılır.

Asimetrik (açık anahtarlı) şifrelemenin dezavantajları

- Kullanılan anahtarların matematiksel olarak daha kompleks işlemler içermesi sebebiyle simetrik şifrelemeye göre daha yavaştır.
- Sistemin fazla CPU harcamasına neden olur.

3.2.2. Gizli anahtarlı (simetrik) sistemler

Açık anahtarlı (asimetrik) sistemlerin aksine, gizli anahtarlı (simetrik) sistemlerde iletinin şifrelenmesi ve deşifrelenmesinde kullanılan anahtarlar aynıdır. Bu anahtarlara gizli anahtar (secret) denir. Bu anahtar türü, yalnızca mesajlaşmak isteyen kişilerce bilinmelidir ve başkalarının (3. Kişilerin / art niyetli) eline geçmemesi gerekir.

Simetrik anahtarlı sistemler işleyiş bakımından asimetrik anahtarlara nazaran daha basit ve daha hızlıdır, ancak dezavantajı her iki tarafın anahtarı güvenli bir şekilde değiştirmesi zorunluluğudur. Anahtar elinde olmayan birisi şifrelenmiş mesaja ulaşabilese de bu mesajı deşifre etmesi mümkün değildir. Simetrik anahtar şifrelemesi bazen gizli anahtar şifrelemesi veya özel anahtar şifrelemesi olarak da adlandırılır. Simetrik anahtar şifrelemesinde, tek anahtar; şifreleme ve şifre çözme işlemi için kullanılır, yani aynı anahtar verilerini kullanarak şifrelenebilir ve şifresi çözülür (Aghayev, 2017).

Simetrik anahtarlara örnek olarak AES,DES, 3DES, Blowfish, IDEA, RC4 ve SAFER verilebilir (Alkan ve Sağıroğlu, 2005).

Şekil 3.2. Simetrik Şifreleme (Frolov, 2021)

Simetrik (kapalı anahtarlı) şifrelemenin avantajları

- Asimetrik şifrelemeye göre daha hızlıdır.
- Donanımla beraber kullanılabilirler.
- Kullanımları daha pratik ve kolaydır.
- Güçlü şifrelerin oluşturulmasında aracı olarak kullanılabilirler.

Simetrik (kapalı anahtarlı) şifrelemenin dezavantajları

- Güvenli anahtar dağıtımı zordur.
- Kapasite sıkıntısı vardır.
- Kimlik doğrulama işlevi görmez. Anahtara sahip herkes olaya dahil olur.
- Bütünlük sağlaması yoktur. Anahtarı ele geçiren kişi veriyi değiştirmiş olabilir.

Açık ve gizli anahtarlı sistemlerin karşılaştırılması

- Simetrik (kapalı) anahtarlı şifrelemenin anahtar uzunluğu, asimetrik (açık) anahtarlı şifrelemeye kıyasla daha kısadır.
- Açık anahtarlı şifrelemenin işlem hızı, simetrik anahtarlı şifrelemeye göre daha düşük düzeyde gerçekleşir.
- Simetrik anahtarlı şifrelemede iletişim için gerekli olan anahtarı her iki tarafında bilip gizli tutması gerekirken, asimetrik anahtarlı şifrelemede ise sadece kendi özel anahtarlarını gizli tutması yeterlidir.
- Simetrik anahtarlı şifrelemede güvenlik açısından anahtarın sürekli değişmesi gerekirken, asimetrik şifrelemede şahsi veya özel anahtarların uzun süreli kullanımı mümkündür.

- Açık anahtar yapılarının genelinde, simetrik anahtar yapılarına göre daha verimli dijital imza mekanizmaları elde edilir.

Simetrik ve asimetrik algoritmaları güvenlik unsuru açısından incelemek için aşağıdaki tabloyu incelemek yararlı olacaktır (Başkök, 2007).

UNSURLAR	SİMETRİK SİSTEMLER	ASİMETRİK SİSTEMLER
Gizlilik	Sağlar	Sağlar
Bütünlük	Sağlar	Sağlar
Kimlik doğrulama	-	Sağlar
İnkar edilememezlik	-	Sağlar
Hesaplama hızı	Yüksek	Düşük
Güvenlik	Anahtar uzunluğuna bağlı	Anahtar uzunluğuna bağlı
Genel değerlendirme	Mesaj şifrelemede kullanılması iyi sonuç vermektedir.	Anahtar şifrelemede kullanılması iyi sonuç vermektedir.

Tablo 3.1 Kripto Sistemlerin Karşılaştırılması

3.2.3. Hibrit Sistemler

Daha önce yukarıda bahsedilen simetrik ve asimetrik sistemlerin birbirine kıyasla artı ve eksi yönleri vardır. Bu iki sistemin artı yönlerinden yola çıkarak geliştirilen yeni sisteme hibrit sistem adı verilmektedir. Simetrik sistemlerde temel problem; anahtarın karşı tarafa güvenli bir şekilde aktarımıdır (Mesajlaşmada tek anahtar kullanıldığı için).

Asimetrik şifreleme ise simetrik şifreleme algoritmalarına göre daha zayıftır. İşte tam da buradan hareketle hibrit sistemler, iletinin (mesajın) simetrik bir algoritma ile şifrelenmesi ve bu algoritmada kullanılan anahtarın da asimetrik bir algoritma ile şifrelenip, gönderilecek ileti ile birlikte iletilmesi ilkesine dayanır. Böylece iletinin şifrelenmesi, simetrik algoritmadan dolayı hızlı olup, anahtarın iletimi de, asimetrik algoritmadan dolayı güvenli olur. Bu yapıların beraber kullanılması ile hız ve güvenirlik bir arada sağlanabileceğinden, verimlilik artacaktır.

Bu sebeple, birçok işlemde bu ve buna benzer yaklaşımlar tercih edilmekte ve geliştirilmektedir (Alkan ve Sağıroğlu, 2005). PGP ve GnuPG gibi yazılımlar bu sisteme örnek olarak gösterilebilir.

3.3. Klasik Kriptoloji

Bu bölümde ele alınan şifreleme sistemleri incelenirken; Stinson (1995), Menezes ve *ark.,* (1997), Buluş (2006), Şiap (2008), Babaoğlu (2009) ve Külen (2013) kaynaklarından yararlanılmıştır.

Bu bölümde simetrik ve asimetrik şifreleme sistemlerinden bazıları incelenecektir. İlgili kriptosistemin şifrelenişini ve şifre çözme fonksiyonlarını verip, bazılarının kriptanalizinin nasıl yapıldığı anlatılmaya çalışılacaktır.

Çalışmanın başında da ifade edildiği gibi konunun anlaşılırlığının kolay olması açısından olay basitleştirilecek olursa; göndericinin Ferhat, alıcının ise Taha olduğunu ve iletiye ulaşmak isteyen 3. kişinin Berat olduğu varsayılsın.

Ferhat'ın Taha'ya ulaştırmak istediği ileti bir metin, nümerik bir veri veya yapısı keyfe bağlı herhangi bir şey olabilir. Ferhat kendisine ait anahtar ile iletiyi şifreleyip Taha'ya gönderir. Berat bu iletişim esnasında şifreli iletiyi gözlemler ancak çözümleyemez. Taha aldığı şifreli iletiyi kendisine ait olan anahtar ile çözümler ve düz metni elde eder.

Şifreleme sisteminde kullanılan bazı kavramlar, matematiksel olarak aşağıdaki gibi yeniden tanımlanabilir.

Tanım 3.3.1. Bir kriptosistem aşağıdaki koşulları sağlayan bir sıralı beşliden (P, C, κ, E, D) oluşur.

(i) P; Düz metinlerin sonlu kümesidir.

(ii) C; Şifreli metinlerin sonlu kümesidir.

(iii) κ; Muhtemel anahtarların sonlu kümesidir.

(iv) $E = E_k : k \in \kappa$ şifreleme ve $D = \{D_k : k \in \kappa\}$ şifre çözme fonksiyonlarının kümesidir.

(v) Her $k \in \kappa$ için bir şifreleme fonksiyonu $E_k \in E$ ve buna bağlı olarak bir şifre çözme fonksiyonu $D_k \in D$ vardır. Burada $E_k : P \to C$ ve $D_k : C \to P$

fonksiyonları, her $x \in P$ için $D_k\, E_k\, x = x$ eşitliğini sağlarlar (Stinson,1995).

Burada son husus olarak; E_k kullanılarak şifrelenen düz metin D_k kullanılarak deşifrelenip orijinal düz (asıl metin) metnin elde edilebileceğini izah etmektedir. Sistemin işleyişi özetlenecek olursa; Ferhat, açık ağlar üzerinden Taha'ya bir mesaj göndermek istesin. Öncelikle rastgele bir anahtar ($k \in \mathcal{K}$) seçer. Bu işlem ya her ikisi aynı yerdeyken yapılır ya da güvenli bir kanal üzerinden anahtar kararlaştırılır ve iletimi yapılır.

İletilecek mesaj ilk olarak nümerik hale getirilir ve bir sayı dizisi elde edilir. Dolayısıyla P, C ve κ ' yı doğal sayıların alt kümeleri olarak değerlendirmek genelliği bozmaz. Bu sayı dizisinin $n \geq 1$ ve $1 \leq i \leq n$ olmak üzere $x_i \in P$ için, $x = x_1 x_2 \dots x_n$, şeklinde olduğunu varsayalım. Akabinde Ferhat $1 \leq i \leq n$ için $y_i = E_k x_i$ formülü ile her bir x_i düz metnine karşılık bir y_i değeri bulur ve $y = y_1 y_2 \dots y_n$ sayı dizisini elde eder ki bu şifreli metnin nümerik karşılığıdır.

Taha $y = y_1 y_2 \dots y_n$ şifreli metni alıp $x_i = D_k y_i$ şifre çözme fonksiyonu ile orijinal $x = x_1 x_2 \dots x_n$ düz metne karşılık gelen nümerik değerleri elde eder. Buradan, hem (D_k) hem de (E_k)'nın bire-bir fonksiyon olduğunu belirtmek gerekir. Aksi takdirde, şifre çözülünce orijinal metin elde edilemez.

Öte yandan, eğer $P = C$ eşitliği sağlanıyorsa, şifreleme fonksiyonu bir permütasyondur. Yani, eğer düz metinler ile şifreli metinlerin kümesi özdeş ise, şifreleme fonksiyonu bu kümenin elemanlarının yeni bir sıralanışını verir (Külen, 2013).

3.3.1 Kaydırma şifresi (The Shift Cipher)

Bu kısımda, temeli modüler aritmetiğe dayanan kaydırma şifresinin tanımı yapılıp, örneklendirilecektir.

Tanım 3.3.1.1 Kaydırma Şifresi (shift cipher)

P=C=K=Z_{29} olsun, $0 \leq k \leq 28$ için,

$e_k(x) = (x + k)\ mod\ 29$

ve

$d_k(y) = (y - k)\ mod\ 29$

$(x, y) \in Z_{29}$ olarak tanımlanır.

Kaydırma şifresi kullanarak şifreleme yapmak için Türk alfabesi kullanılacaktır. Alfabemizde 29 harf bulunduğundan $mod\ 29$ 'a göre işlem yapılacaktır. Örneklerde kolaylık olması açısından bu durumu aşağıdaki tablo ile görselleştirmek daha faydalı olacaktır. (Özyılmaz, 2014).

A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z
15	16	17	18	19	20	21	22	23	24	25	26	27	28

Tablo 3.2. Türk Alfabesinin mod29'a göre sayı karşılıkları

Örnek 3.3.1.1.

Kaydırma şifresi için $k = 8$ ve düz metin de ''PALANDÖKENDAĞI'' olsun. İlk olarak 1. Satıra metni, 2. Satıra Tablo 3.2.' e göre metnin tamsayı karşılığı yazılır, 3. Satıra ise $k = 8$ olduğu için bu tamsayıların 8 fazlasının $(mod29)$ a göre dengini yazılır.

P	A	L	A	N	D	Ö	K	E	N	D	A	Ğ	I
19	0	14	0	16	4	18	13	5	16	4	0	8	10
27	8	22	8	24	12	26	21	13	24	12	2	16	18

Sonra 2. Satırda oluşan tamsayıların hangi harfe karşılık geldiğini Tablo 3.2.'den bulup yazıldığında şifreli metnin ''YĞŞĞUJVSKUJCNÖ'' olduğu görülür.

Bu şifreli metni okunaklı hale (orijinal) getirmek için;

$$D_k\ y = y - 8(mod29)$$

fonksiyonu kullanılır.

Kaydırma şifresi çokta güvenilir bir metot değildir. Çünkü ayrıntılı arama yöntemiyle çözümlenmesi uzun sürmeyecektir. Kendi alfabemiz düşünüldüğünde en fazla 29 deneme ile şifreli bir metin anlamlı hale getirilebilir. Bu durum aşağıdaki örnekle izah edilebilir.

Örnek 3.3.2. Şifreli mesaj '' TRORCHÜTNÜMKIÜP'' olsun. Şimdi bu şifreli mesaj, olası kaydırma şifre anahtarları kullanılarak deşifre edilmeye ve anlamlı hale getirilmeye çalışılırsa aşağıdaki durum gerçekleşir.

$k = 1$ için USÖSÇIVUOVNLİVR
$k = 2$ için ÜŞPŞDİYÜÖYOMJYS
$k = 3$ için VTRTEJZVPZÖNKZŞ
$k = 4$ için YUSUFKAYRAPOLAT

Bu örnekte de görüldüğü üzere 4. Adımda şifreli metin anlamlı hale getirilmiştir. Öyleyse anahtar $k = 4$ 'tür. Buradan da anlaşılacağı üzere bir kriptosistemin güvenilir olması için çözümünün ya çok zor ya da mümkün olmaması gerekir. Ancak bu şifreleme bize bunu vaat etmiyor (Külen, 2013).

Tarihte bu şifreleme türü oldukça kullanılmıştır. Hatta en meşhurlarından biri, Büyük Roma İmparatoru Julius Caesar'ın komutanlarıyla haberleşirken kullandığı sistemdir. Caesar şifrelemesinde, harfleri hep 3 öteleyerek şifreleme yapmıştır. Yani kaydırma şifresi $k = 3$ için Sezar şifresi de denilebilir (Stinson, 2002).

3.3.2. Yer Değiştirme Şifresi (The Substition Cipher)

$P = C = Z_{29}$ olmak üzere, $\mathcal{K}; 29$ ifadenin yani $0,1,\ldots,28$'in tüm olası permütasyonlarından meydana gelsin. Her $\pi \in \mathcal{K}$ permütasyonu için; π^{-1} , π ' nin ters permütasyonu olmak üzere ;

$$e_\pi(x) = \pi(x)$$

ve

$$d_\pi(y) = \pi^{-1}(y)$$

olarak tanımlanır (Stinson, 2002).

Düz metnin (açık metin, anlamlı metin) karakterleri büyük harflerle, şifreli metnin (kapalı metin) karakterleri ise küçük harflerle aşağıdaki Tablo 3.3'de gösterilmiştir.

A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L
c	a	z	u	v	ğ	ç	y	ü	b	d	h	s	ı	j

M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z
i	k	n	e	o	p	g	ö	ş	f	m	t	l	r

Tablo 3.3. Örnek π fonksiyonu

Şifreli metni anlamlı hale getirme fonksiyonu ise alfabetik sıranın tersidir. Yine bu durum da aşağıdaki Tablo 3.4'te gösterilmiştir.

a	b	c	ç	d	e	f	g	ğ	h	ı	i	j	k	l
B	H	A	F	I	O	U	S	E	İ	K	M	L	N	Y

m	n	o	ö	p	r	s	ş	t	u	ü	v	y	z
U	O	P	Ş	R	Z	J	T	V	Ç	Ğ	D	G	C

Tablo 3.4. Örnek π^{-1} fonksiyonu

Örneklendirilecek olursa ‘’lcpdkşnojckşdtcp’’şifreli metni, bu fonksiyon kullanılarak çözümlendiğinde ‘’YARIN TOPLANTI VAR’’ açık metni elde edilmiş olur.

3.3.3. Affine Şifreleme Sistemi (Affine Cipher)

Bu şifreleme metodunun en bariz özelliği şifreleme işleminde $y = ax + b$ doğrusal fonksiyonunun kullanılmasıdır. Buradaki x değişkeni şifrelenecek mesajı, y değişkeni de şifrelenmiş mesajı temsil etmekte olup $(a\,, b)$ sıralı ikilisi anahtarı oluşturmaktadır. Bu metot, matematiksel işlemlerle aşağıda izah edilmeye çalışıldı.

Tanım 3.3.1. de verilen P, C, κ, E, D sıralı beşlisinde, n kullanılan dilin harf sayısını göstermek üzere;

$$P = C = Z_n \text{ ve } \kappa = \{a, b : a, b \in Z_n\} \text{ olsun.}$$

$$a, b \in Z_n \quad \text{ve} \quad ebob(a, n) = 1$$

olmak üzere şifreleme fonksiyonu;

$$E_{a,b}x \equiv ax + b \ (mod\ n)$$

şeklinde tanımlanır. Burada özel bir durum söz konusudur. $a = 1$ alındığında fonksiyon kaydırma şifresine dönüşür. Bu sistemde şifreleme yapılabilmesi için gerekli şart, Affine fonksiyonunun bire-bir olması durumudur. Bire-bir olması için de gerek ve yeter şart, her $y \in Z_n$ için, $ax + b \equiv y \ (mod\ n)$ denkleminin x için tek bir çözümü olması gerekir. Bu denklem $ax \equiv y - b \ (mod\ n)$ denklemine denktir. Bu denklemin tek çözümünün olması için gerek ve yeter şart $ebob(a, n) = 1$ olmasıdır. Şimdi $ebob(a, n) = 1$ olduğunu varsayalım. Bu durumda a’nın $(mod\ n)$ tersi vardır ve

$$x \equiv a^{-1}.\ y - b \ (mod\ n)$$

olur. Böylece $\kappa = (a, b) \in Z_n x Z_n,\ ebob\ (a, n) = 1\ ve\ k = (a, b)\ \in\ \kappa$ olmak üzere; her $x \in Z_n$ için;

$$E_k\, x = ax + b\ (mod\ n)$$

şifreleme ve her $y \in Z_n$ için;

$$D_k\, y = a^{-1}\, y - b\ (mod\ n)$$

şifre çözme fonksiyonları elde edilir (Külen, 2013).

Yukarıda izahı yapılan şifreleme, en sade haliyle aşağıda örneklendirilmeye çalışılmıştır.

Örnek 3.3.3.1. ‘’DADAŞ’’ sözcüğü şifrelenmek istenilirse öncelikle harf karşılıkları yazılır.

D= 4 , A= 0 , Ş= 22

Sonra anahtar değeri de rastgele (3,4) olarak belirlenir.

$$E(D) = 3.4 + 4\ (mod 29) = 16 \rightarrow N$$
$$E(A) = 3.0 + 4\ (mod 29) = 4 \rightarrow D$$
$$E(Ş) = 3.22 + 4\ (mod 29) = 12 \rightarrow J$$

olur. O halde şifreli metin ‘’ $NDNDJ$ ‘’ olmuş olur.

Şimdi bu sözcüğü tekrar orijinal hale getirelim.

Önce z değerini bulalım.

$3.z\ (mod\ 29) = 1$ için z değeri 10 olarak bulunur (z = 10).

$$N = 16\,, \qquad D = 4\,, \quad J = 12$$
$$E'(N) = 10.(16 - 4)\ mod\ 29 = 4 \rightarrow D$$
$$E'(D) = 10.(4 - 4)\ mod\ 29 = 0 \rightarrow A$$
$$E'(J) = 10(12 - 4)\ mod\ 29 = 22 \rightarrow Ş$$

Böylece deşifrelenmiş hali '' $NDNDJ$ '' olan metin, orijinal hali olan ''DADAŞ'' haline çevrilmiş oldu.

3.3.4. Vigenere şifresi

Bu şifreleme yöntemini ilk olarak Giovan Batista Belasa tanıtmış ve kullanmış olsa da, yöntemi düzenleyip geliştiren kişi Blaise De Vigenere olduğu için sistemin ismi 'Vigenere şifresi' olarak anılır. Bu sistemde şifrelenecek karakterler birebir değiştirilmez. Onun yerine anahtar uzunluğunda bir metnin bir algoritma ile aynı anda şifrelenmesi tercih edilir. Buna çoklu alfabetik (polyalphabetic) adı verilir. Bu yöntem matematiksel olarak şöyle izah edilebilir.

m sabit pozitif bir tamsayı ve n şifreleme yapılacak dilin karakter sayısı olmak üzere;

$$P = C = \kappa = (Z_n)^m \text{ olsun.}$$

Bir $k = k_1,\ k_2, \dots\dots, k_m$ anahtarı için şifreleme fonksiyonu;

$$E_k x_1, \dots . x_m = x_1 + k_1, \dots\dots, x_m + k_m \ (mod\ n)$$

ve şifre çözme fonksiyonu;

$D_k y_1, \dots\dots y_m = y_1 - k_1, \dots . y_m - k_m \ (mod\ n)$

şeklinde tanımlanır.

Şifreleme işlemi yapılırken, ilk olarak düz metin anahtar uzunluğu olan m uzunluklu parçalara bölünür. Bölünen parçalar, k anahtarı ile şifreleme fonksiyonu kullanılarak şifrelenir ve sonrasında m tane alfabetik karakter bir anda şifrelenir. Yine şifre çözme işlemi de aynı şekilde yapılır. Örneklendirmek istenilirse;

Örnek 3.3.4.1.

''CEBİRBİLİMİ'' açık metni bu yöntemle şifrelenmek istensin. Anahtar kelime "akıl" olarak belirlenirse anahtar uzunluğu $m = 4$ olmuş olup, yine bu anahtar kelimemizin nümerik eşdeğeri de çizelgeden bakıldığında k =(0,13,10,14) olarak elde edilir. Önce düz metnin nümerik karşılığı, hemen onun altına ise şifrenin nümerik değerlerini yazıp mod29' a göre toplandığında elde edilen sayıların harf karşılıkları metnin şifrelenmiş halini vermiş olur.

AÇIK METİN	C	E	B	İ	R	B	İ	L	İ	M	İ
AÇIK METİN	2	5	1	11	20	1	11	14	11	15	11
ANAHTAR	0	13	10	14	0	13	10	14	0	13	10
ŞİFRELİ METİN	2	18	11	25	20	14	21	28	11	28	21
ŞİFRELİ METİN	C	Ö	İ	Ü	R	L	S	Z	İ	Z	S

Tablo 3.5. Örnek Vigenere Şifrelemesi

En sonda da görüldüğü gibi 'CEBİRBİLİMİ' açık metni ' CÖİÜRLSZİZS' olarak vigenere şifreleme yöntemiyle şifrelenmiş oldu. Şifreli metin tekrar asıl metne çevrilmek istenilirse aynı anahtarı kullanır ancak toplama yerine çıkarma işlemi yapılması gerekir. Tablo ile gösterilmek istenilirse aşağıdaki gibi olur.

ŞİFRELİ METİN	C	Ö	İ	Ü	R	L	S	Z	İ	Z	S
ŞİFRELİ METİN	2	18	11	25	20	14	21	28	11	28	21
ANAHTAR	0	13	10	14	0	13	10	14	0	13	10
AÇIK METİN	2	5	1	11	20	1	11	14	11	15	11
AÇIK METİN	C	E	B	İ	R	B	İ	L	İ	M	İ

Tablo 3.6. Örnek Vigenere Deşifrelemesi

Vigenere şifreleme sisteminin kriptanalizi: m uzunluğundaki anahtar kelimelerin sayısı n^m tanedir. Detaylı anahtar inceleme yöntemiyle deşifreleme yapılabilir. Tek tek bakmak güç olsada günümüz bilgisayarları sayesinde çokta zor bir durum değildir. Ancak yine de mono alfabetik kriptosistemlere göre çok daha kuvvetlidir. Ayrıca anahtar kelimenin boyu arttıkça şifreleme daha güç olmaktadır (Külen, 2013).

3.3.5. Hill Şifrelemesi (Blok Şifreleme)

Simetrik anahtar blok şifreler kriptografik sistemlerin en elzem ve en iyi bilinen elemanlarıdır. Bir blok şifre açık metni çözebilen ve aynı anda bir bloğu şifreleyen şifreleme sistemidir. Diğer bir deyişle bir blok şifre m- bitlik şifreli metin bloğuna dönüştüren bir fonksiyondur (Soyalıç, 2004).

Tanım 3.3.5.1. Hill (Blok) Şifreleme

$m \geq 2$ olan bir tamsayı olsun. $P = C = (Z_{29})^m$ ve
$K = \{Z_{29}'da\ m\ x\ m$ tipinde tersi alınabilen matrisler$\}$ olsun. Bir k anahtarı için ;

$$e_k(x) = x.k$$

ve

$$d_k(y) = y.k^{-1}$$

olarak tanımlanır (Stinson, 2002).

Örnek 3.3.5.1. ‘’ADALET’’ açık metni, $\begin{bmatrix}1 & 2 & 4\\3 & 1 & 0\\2 & 5 & 3\end{bmatrix}$ anahtar matrisi ile şifrelenmek istensin.

Anahtar matris $3\ x\ 3$ türünde olduğundan $m = 3$ ve Türkçe karakterler kullanıldığından $n = 29$ olur. Yapılacak olan şifrelemeler üçerli gruplar halinde yapılır. Şifrelenecek metnin nümerik değerleri Tablo 3.2.‘den bakıldığında;

(A,D,A) = (0, 4 ,0) ve (L , E , T) = (14 , 5 , 23)

vektörleri olur. Bu değerler;

$$E_k x = y_1, y_2, y_3 \equiv (x_1, x_2, x_3)\begin{bmatrix}1 & 2 & 4\\3 & 1 & 0\\2 & 5 & 3\end{bmatrix} \qquad (mod\ 29)$$

Şifreleme fonksiyonunda yerine yazılır. Böylece,

$$y_1, y_2, y_3 \equiv (\ 0\ ,4\ ,0\ )\ \begin{bmatrix}1 & 2 & 4\\3 & 1 & 0\\2 & 5 & 3\end{bmatrix} \equiv (12\ ,4\ ,0\ )\ mod\ 29$$

$$y_1, y_2, y_3 \equiv (\ 14\ ,5\ ,23\ ) \qquad \begin{bmatrix}1 & 2 & 4\\3 & 1 & 0\\2 & 5 & 3\end{bmatrix} \equiv (\ 17\ ,3\ ,9\ )\ mod\ 29$$

değerleri elde edilir ve şifreli metin ‘’JDAOÇH’’ olmuş olur.

3.3.6 Permütasyon Şifresi

Şimdiye kadar görülen tüm kriptosistemlerde kaydırma işlemi yapılarak şifreleme yapılıyordu. Böylelikle asıl metin, kaydırma işlemi sayesinde

anlamsız(şifreli) metin haline gelmiş oluyordu. Permütasyon şifresinde ise harfleri kaydırma (öteleme) yapılmaz, düz metin karakterleri değiştirilmez; ancak, tekrardan düzenlenerek yerleri değiştirilir (Tuncal, 2008).

Tanım 3.3.6.1. Permütasyon Şifresi

m rastgele bir pozitif tamsayı olsun. $P = C = (Z_{29})^m$ ve K , $\{1, \dots, m\}$ 'in tüm permutasyonlarından oluşsun. Bir π anahtarı için ;

$$e_\pi(x_1, \dots . x_m) = \big(x_\pi(1), \dots . x_\pi(m)\big)$$

$$d_\pi(y_1, \dots y_m) = (y_{\pi^{-1}}(1), \dots , y_{\pi^{-1}}(m))$$

Buradaki π^{-1} , π permutasyonunun tersidir (Stinson, 2002).

Örnek 3.3.6.1. $m = 5$ ve anahtar ise aşağıdaki gibi π permutasyonu olsun.

x	1	2	3	4	5
$\pi(x)$	3	5	1	2	4

Son durumda π permutasyonunun tersi olan π^{-1} permutasyonu ise $m = 5$ için;

x	1	2	3	4	5
$\pi^{-1}(x)$	3	4	1	5	2

olur. Şimdi buradan hareketle şifrelemek istenilen mesaj ‘’TÜMİYİŞEYLERSABIRDANSONRAGELİR’’ olsun. Burada $m = 5$ olduğu için mesaj 5'li bloklara ayrılıp, bu 5'li bloklar π permutasyonuna göre düzenlensin.

TÜMİY | İŞEYL | ERSAB | IRDAN | SONRA | GELİR

5'li bloklara ayrıldıktan sonra π permutasyonuna (1. Bloğu baz alarak olay kısaca özetlenmeye çalışılırsa; önce 1. harf olan T için 3. harf M, 2. harf Ü için 5. harf Y, 3. harf M için 1. harf T, 4. harf İ için 2. harf Ü ve son olarak 5. harf Y için 4. harf olan İ yerine yazıldığında ‘’MYTÜİ’’ şifreli bloğu elde edilmiş olur.) göre düzenlenirse;

MYTÜİ | ELİŞY | SBERA | DNIRA | NASOR | LRGEİ

olur. Böylece şifreli metin;
‘’MYTÜİELİŞYSBERADNIRANASORLRGEİ’’ olmuş olur. Buradaki şifreli metin deşifreleme yoluyla açık metin haline getirilebilir. Dolayısıyla deşifreleme yapılacağı için burada π^{-1} permutasyonu kullanılacaktır. Şifreli metin ;

MYTÜİ | ELİŞY | SBERA | DNIRA | NASOR | LRGEİ

idi. π^{-1} permutasyonuna göre gerekli işlemler yapıldığında;

TÜMİY | İŞEYL | ERSAB | IRDAN | SONRA | GELİR

düz metni elde edilir. Yani ‘’TÜMİYİŞEYLERSABIRDANSONRAGELİR’’ ana metnine ulaşılır.

3.3.7 Akış Şifreleri

Simetrik anahtar şifreleme tasarılarının en önemli sınıflarından birini oluşturur. Birbirine eş uzunlukta bloklara sahip çok basit blok şifreler oldukları da ifade edilebilir (Soyalıç, 2004).

Tanım 3.3.7.1. Akış Şifreleri (Stream Ciphers)

Akış şifresi g fonksiyonu ve (P,C,$\mathcal{K}$,L,E,D) öğeleri ile birlikte aşağıdaki şartları sağlamalıdır:

1. P , mümkün olan bütün açık metinleri bulunduran sonlu küme
2. C , mümkün olan bütün şifre metinleri bulunduran sonlu küme
3. $\mathcal{K}$, mümkün olan bütün anahtarları bulunduran sonlu küme
4. L , anahtar akış alfabesi (keystream alphabet) olarak adlandırılan sonlu küme

5. *g;* anahtar akış üretecidir (keystream generator), girdi olarak bir $\mathcal{K}$ anahtarını alır ve her $i \geq 1$ için $z_i \in L$ olmak üzere anahtar akışı olarak adlandırılan z_1..... sonsuz dizisini üretir.

6. Her $z \in L$ için, $e_z \in E$ olan bir şifreleme kuralı ve $d_z(e_z(x)) = x$ olan bir deşifreleme kuralı vardır. $e_z: P \to C$ ve $d_z: C \to P$ açık metindeki her $x \in P$ için $d_z(e_z(x)) = x$ olacak şekilde fonksiyonlardır (Stinson, 2002).

En iyi bilinen akış şifresi Vernam Şifresidir.

4. KRİPTOSİSTEMLER

3. bölümde açık anahtarlı (asimetrik) sistemlerden, gizli anahtarlı (simetrik) sistemlerden ve bunların birbiri ile kıyasından yola çıkarak artı ve eksi yönlerinden bahsedilmişti. Bu bölümde ise en çok kullanılan açık anahtar(asimetrik) algoritmalarından ve gizli anahtar (simetrik) algoritmalarından bahsedilip, ardından anahtar dağıtım şemaları anlatılacaktır.

4.1. En Çok Kullanılan Gizli Anahtarlı (Simetrik) Kriptografi Algoritmaları

Bu şifreleme algoritmasında şifreleme ve deşifreleme algoritmalarının çalışma prensipleri birbirinin tersi şeklinde olur. İlk olarak haberleşecek taraflar arasında ortak bir gizli anahtar tayin edilir ve haberleşme bu ortak anahtar üzerinden yürütülür. Tabi bu haberleşme esnasında tek anahtar olduğu için bu anahtarın güvenliği çok önemlidir. Çünkü haberleşen taraflar arasında birinden diğerine iletim esnasında anahtarın 3. kişilerin eline geçme ihtimali söz konusudur.

Geleneksel simetrik şifreleme algoritmaları ilk olarak 1973'te IBM'de çalışan Hort Feistel tarafından geliştirilen Feistel Network'una dayanır. Aşağıdaki Şekil 4.1. de gösterilen bu algoritmada 2w bit uzunluğunda olan şifresiz metin, iki eş parça olmak üzere sağ ve sol parçalara ayrılır. Sonrasında her bir seferde ana şifre tarafından üretilen alt şifre ile sağ tarafa F fonksiyonu uygulanır. Bunun neticesinde sol taraf ile EXOR mantıksal işlemine tabi tutulur. Akabinde elde edilen sonuçlar çaprazlanır. Çaprazlama neticesinde sağ taraf sola, sol taraf sağa geçer. Turlar bu şekilde devam eder. Asıl anahtardan alt anahtarlar, her turda üretilerek F fonksiyonuna girdi olarak kullanılır. Feistel algoritmasının temel karakteristik özellikleri aşağıda açıklanmıştır (Tuncal, 2008).

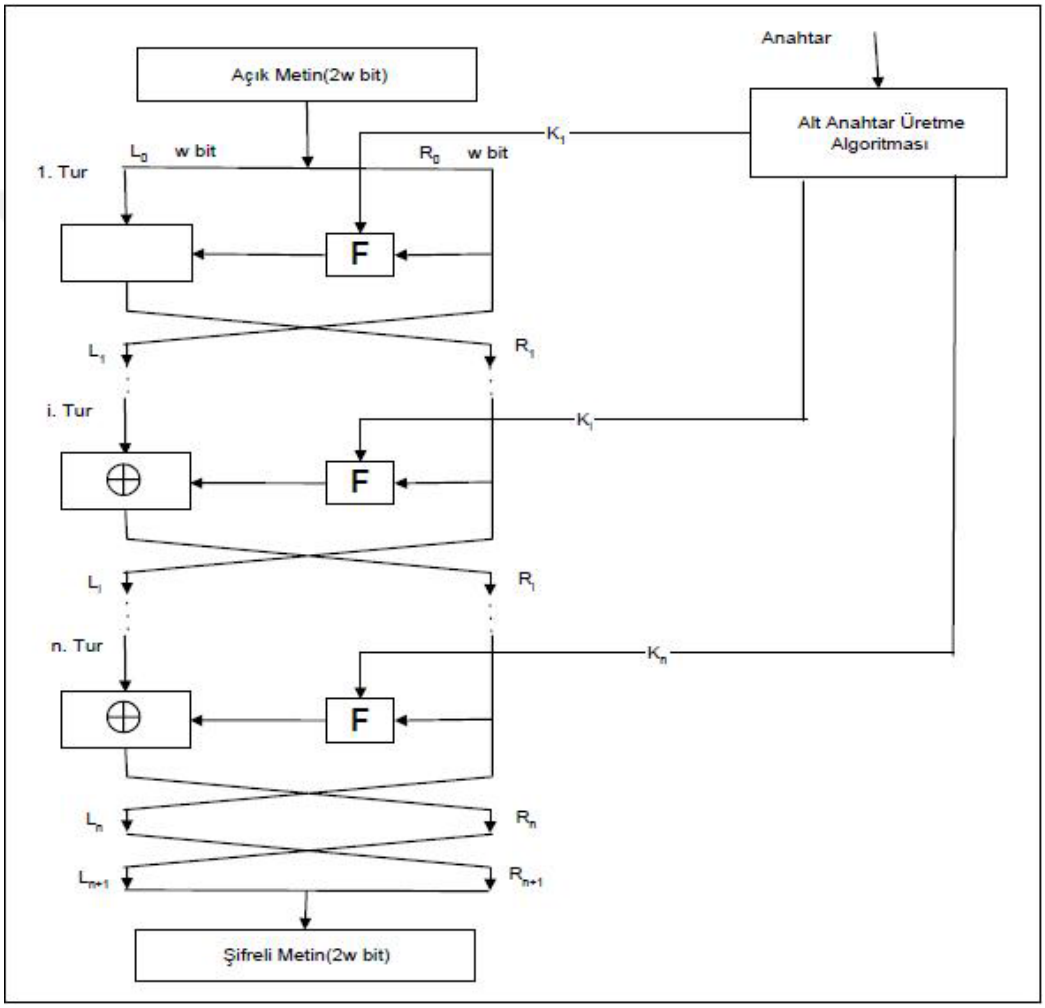
Blok Uzunluğu: Blok uzunluğunun fazla olması daha güvenli olması anlamına gelir ancak şifreleme/deşifreleme hızını yavaşlatır. Çoğunlukla 64 bitlik blok uzunluğu tercih edilir.

Anahtar Uzunluğu: Anahtar uzunluğunun büyük olması yine daha güvenli olduğu anlamına gelir ancak bu da şifreleme/deşifreleme hızını yavaşlatır. En çok tercih edilen anahtar uzunluğu 128 bittir.

Tur Sayısı: Tur sayısının fazla olması şifreleme güvenliğini arttırır. Çoğunlukla 16 tur tercih edilir.

Alt Anahtar Üretme Algoritması: Karmaşıklığı yüksek olan tur fonksiyonu kriptoanalizi zorlaştırır.

Tur Fonksiyonu: Tur fonksiyonunun karmaşıklığının çok olması kriptanalizini güçleştirir.



Şekil 4.1. Klasik Feistel Network

Başlıca kullanılan simetrik şifreleme algoritmaları DES, IDEA, Blowfish, RC5, CAST-128 ve AES'dir.

4.1.1. Veri Şifreleme Standardı (Data Encryption Standart (DES))

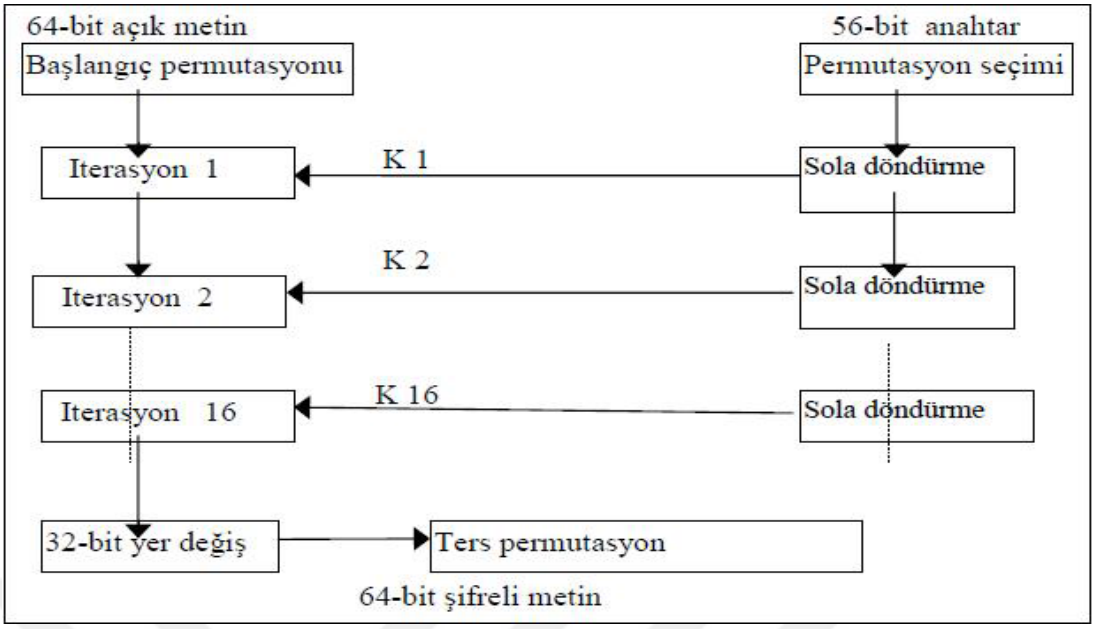
DES; IBM tarafından 1970'lerin başında geliştirilmiş, 1976 yılında Ulusal Güvenlik Ajansına (NSA) sunulmuş ve 1977 yılında yasal olarak atanmıştır. Basit blok şeması Şekil 4.2'de izah edilmiştir. Esası Feistel networküne dayanır. DES algoritması elektronik verilerin şifrelenmesinde kullanılan simetrik şifreleme algoritmasıdır. Günümüz şartlarına göre güvensiz olsa da modern kriptografinin oluşumunda önemli rol oynamıştır.

Daha önce de izah edildiği gibi DES bir blok şifrelemedir ve 64 bit bloklardaki veriyi şifreler. Asıl metnin 64'lük bloğu bir algoritmaya sokulur ve 64 bitlik şifrelenmiş bir metin meydana gelir. Şifrelemede ve şifreyi çözmede aynı algoritma ve anahtarlar kullanılır. Anahtar uzunluğu 56 bittir. Anahtar rastgele bir 56 bit sayı olabilir ve istediğiniz zaman değiştirebilirsiniz.

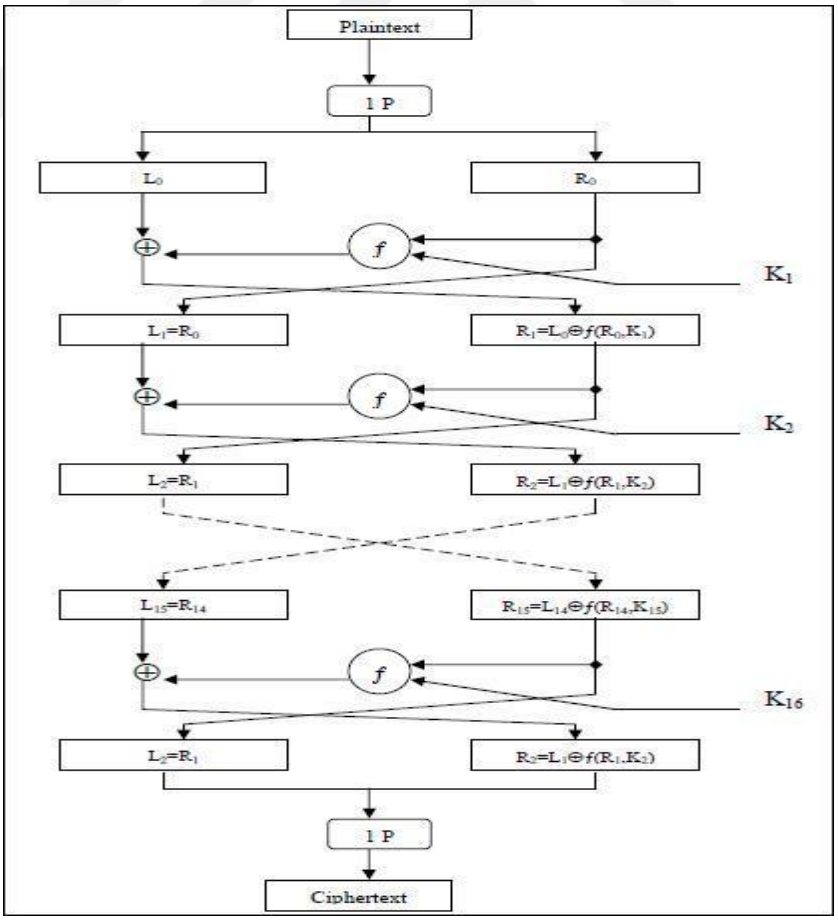
Algoritmanın Özeti

DES algoritması 64 bitlik blok düz metninde işlem görür. Düz metin, ilk permutasyondan sonra yarısı sağda, yarısı solda olmak üzere herbiri 32 bit uzunluğunda iki parçaya bölünür. Daha sonra f fonksiyonu ve anahtar ile birleştirilerek diğer adıma geçilir. Aynı işlem 16 defa tekrarlanır ve 16. turun sonunda, sağ ve sol parçalar bir araya getirilir. Son permutasyondan sonra algoritma tamamlanarak sona erer.

DES algoritmasının genel yapısı (Şekil 4.2.) ve genel işleyişi (Şekil 4.3.) aşağıda gösterilmiştir.



Şekil 4.2. DES Algoritmasının Genel Yapısı



Şekil 4.3. DES Algoritmasının Genel İşleyişi

4.1.2. Uluslararası Veri Şifreleme Algoritması (International Data Encryption Algorithm (IDEA))

IDEA, bir Simetrik blok şifreleme algoritmasıdır ve 1991'de Swiss Federal Institute of Technology'de geliştirilmiştir. Anahtar uzunluğu için 128 bit olması tercih edilir. IDEA, alt anahtar üretim ve tur fonksiyonları açısından DES'ten farklıdır. S-boxes kullanılmaz. XOR, 16 bit tamsayı toplama ve 16 bit tamsayı çarpma matematik işlemlerini kullanır. Kriptanalizi zor ve uğraştırıcı olan bir algoritma türüdür. Alt anahtar üretim algoritması sadece dairesel kaydırma üzerinedir ancak her bir sekiz turda altı alt anahtar üreten kompleks bir yapıya sahiptir. İlk 128 bit anahtar kullanan algoritma olduğu için kriptanalistlerin üzerinde çok çalıştıkları bir algoritma türüdür (İşpınar, 2004).

4.1.3. BlowFish

Blowfish; DES veya IDEA algoritmalarına alternatif olmak ve onların yerini almak için ortaya çıkarılan bir algoritma türüdür. 1993 yılında, o zamana kadar kullanılan şifreleme algoritmalarına alternatif olarak Bruce Schneier tarafından tasarlanmıştır. 32-bit komut işlemcileri baz alınarak tasarlanmıştır. DES'e kıyasla daha hızlıdır.

Bu algoritma türünün patentsiz, lisanssız ve tüm kullanımlar için ücretsiz olması en önemli avantajlarındandır. Blowfish, anahtarların değiştirilmesi haricinde diğer blok şifrelere kıyasla daha hızlıdır. Her bir anahtar, diğer blok şifrelere nazaran çok yavaş olan 4 kilobayt metni şifrelemek için ön işleme gerektirir. Bu durum bazı uygulamalarda kullanımı zorlaştırsa da, SplashID gibi uygulamalarda bu durum engel teşkil etmez. Daha önce de belirtildiği gibi bu algoritma türü herhangi bir patente tabi değildir ve kullanımı ücretsiz olup herkes tarafından kullanılabilir. Bu durum da bu algoritma türünün popülerliğini arttırıyor (Hosseinpour, 2018).

4.1.4. RC5

RC5, diğer algoritmalara nazaran daha değişik özelliklere sahiptir. Blok büyüklüğü (32, 64 veya 128 bit), anahtar büyüklüğü (0 ile 2040 bit) ve tur sayısı (0 ile 255) gibi özelliklere sahiptir. Genelde önerilen parametre türü 64 bit blok, 128

bitlik bir anahtar ve 12 turdur. Bu algoritma türünün önemli özelliklerinden biri veriye bağlı değişkenlerin kullanılmasıdır. RC5; bazı modüler işlemler ve exclusive OR (XOR) içerir. Genel işleyişi Feistel ağına benzerdir. Şifreleme ve şifre çözme işlemleri basit kodlarla ifade edilebilir. Ancak anahtar programı daha karmaşıktır. Bu şifreleme algoritmasının bir diğer ve en önemli özelliklerinden biri de verilerin güncel sistemlerle basitleştirilebilmesidir. Bu da kullanım açısından kriptanalistler için önemli bir özelliktir ve algoritmanın tercih edilme oranını arttırır.

4.1.5. CAST-128

1997'de Entrust Teknolojier'den Carlise Adams ve Stafford Tavares tarafından üzerinde çalışılan ve geliştirilen bir tasarım algoritmasıdır. Herhangi bir algoritma 8 bit artırım ile 40 bitten 128 bite kadar değişiklik gösteren anahtar uzunlukları kullanır. CAST, DES'te kullanılanlara kıyasla daha uzun olan sabit S-Box'ları kullanır. Kriptoanalistçiler için bu S-Box'ların tasarımı son derece önem ihtiva eder. Alt anahtar üretimi diğer algoritmalardan farklıdır. Doğrusal olmayan S-Box'lar kullanılarak alt anahtar üretimi yapılır. Bu algoritmanın önemli özelliklerinden biri de turdan tura değişen F tur fonksiyonudur (Hosseinpour, 2018).

4.1.6. AES

Gelişmiş Şifreleme Standardı olarak anılan AES, ABD hükümetinin gizli bilgileri muhafaza etmek ve bu bilgileri şifrelemek için seçtiği ve dünya çapında yazılım ve donanımda uygulanan simetrik bir blok şifrelemedir. Bu yeni simetrik anahtar, kullanıcıların erişimine açıktır. Bu durum, sunulan tasarımların tam ve şeffaf bir analizine olanak sağlar. NIST (National Institute of Standarts and Technology), yeni gelişmiş şifreleme standardı algoritmasının 128, 192 ve 256 bit ebatlarında tuşları kullanarak 128 bit blokları işleyebilen bir blok şifresi olması gerektiğini ifade etmiştir.

Bundan sonraki çalışmaları, gelişmiş şifreleme standardı algoritması olarak seçilebilmek için gereken diğer önemli hususlar şunlardır:

Güvenlik: Şifrelemede güvenlik en temel unsurlardan biridir. Yabancı unsurlara ve art niyetli kişilerin saldırılarına karşı direnme yeteneği olarak görülen güvenlik unsuru oldukça önemlidir.

Maliyet: Herkesin kullanımına açık olması ve erişiminin kolay olması gibi faktörlerin yanında telif haklarının ücretsiz oluşu algoritmanın kullanımını arttıracaktır.

Uygulama: Kullanılacak olan algoritma türünün esnek olması ve donanım-yazılım uygulamalarındaki kullanışlılığı ve basit olması önemli özelliklerden bir tanesidir.

AES şifreleme algoritması üç tip şifreleme barındırır: Bunlar; AES-128, AES-192 ve AES-256'dir. Her şifre, sırasıyla 128, 192 ve 256 bit şifreleme anahtarlarını kullanarak 128 bitlik bloklardaki verileri şifreler (Hosseinpour, 2018).

4.2. En Çok Kullanılan Açık Anahtar (Asimetrik) Algoritmaları

Açık anahtarlı kriptosistemlerin kullanımı, gizli anahtarlı kriptosistemlere nazaran daha yenidir. Bu algoritma türü ile alakalı ilk öneriyi ortaya atan kişiler, 1976 yılında Diffie-Hellman adlı kişilerdir. Sonrasında 1977 yılında Rivest, Shamir ve Adleman; RSA kriptosistemi adlı yeni bir açık anahtar kriptosistemi ortaya atmışlardır. Bu sürecin akabinde bu alanla alakalı değişik öneri ve çalışmalar ortaya atılmıştır. Bunlardan en mühimi El-Gamal tarafından sunulan El-Gamal Açık Anahtarlı kripto algoritmasıdır.

Daha önce görülen gizli (simetrik) anahtarlı kriptosistemlerde iki tarafın haberleşmesi esnasında tek bir anahtar (gizli anahtar) kullanılıyordu. Bu durum da güvenlik zaafiyeti doğuruyordu. Ancak açık anahtarlı kriptosistemlerde şifreleme ve deşifreleme için farklı anahtar kullanılır. Haberleşecek taraflar arasında mesajı alacak kişi sadece kendisine has olan bir ''gizli anahtar'' ve haberleşeceği kişilere dağıtabileceği bir ''açık anahtar'' çifti belirler. Tabi ki bu iki anahtar birbirinden bağımsız değildir ve bu iki anahtar çifti arasında sıkı bir matematiksel ilişki söz konusudur.

Açık anahtar algoritmalarının 3 temel şeması vardır. Bunlar;

1. Açık Anahtar Dağıtım Şeması (Public-Key Distribution Schemes (PKDS): Bu şema türü, bilginin bazı bölümlerinin güvenli olarak değiştirilmesi için kullanılır.

2. İmza Şeması (Signature Schemes): Bu şema türü de sayısal imza üretmek için kullanılır. Buradaki gizli anahtarın işlevi imzayı üretmek, açık anahtarın işlevi ise doğrulamaktır.

3. Açık Anahtar Şeması (Public Key Schemes (PKS)): Şifrelemek için kullanılır. Burada açık anahtarın fonksiyonu mesajı şifrelemek, gizli anahtarın fonksiyonu ise deşifrelemektir (İşpınar 2004; Tuncal 2008).

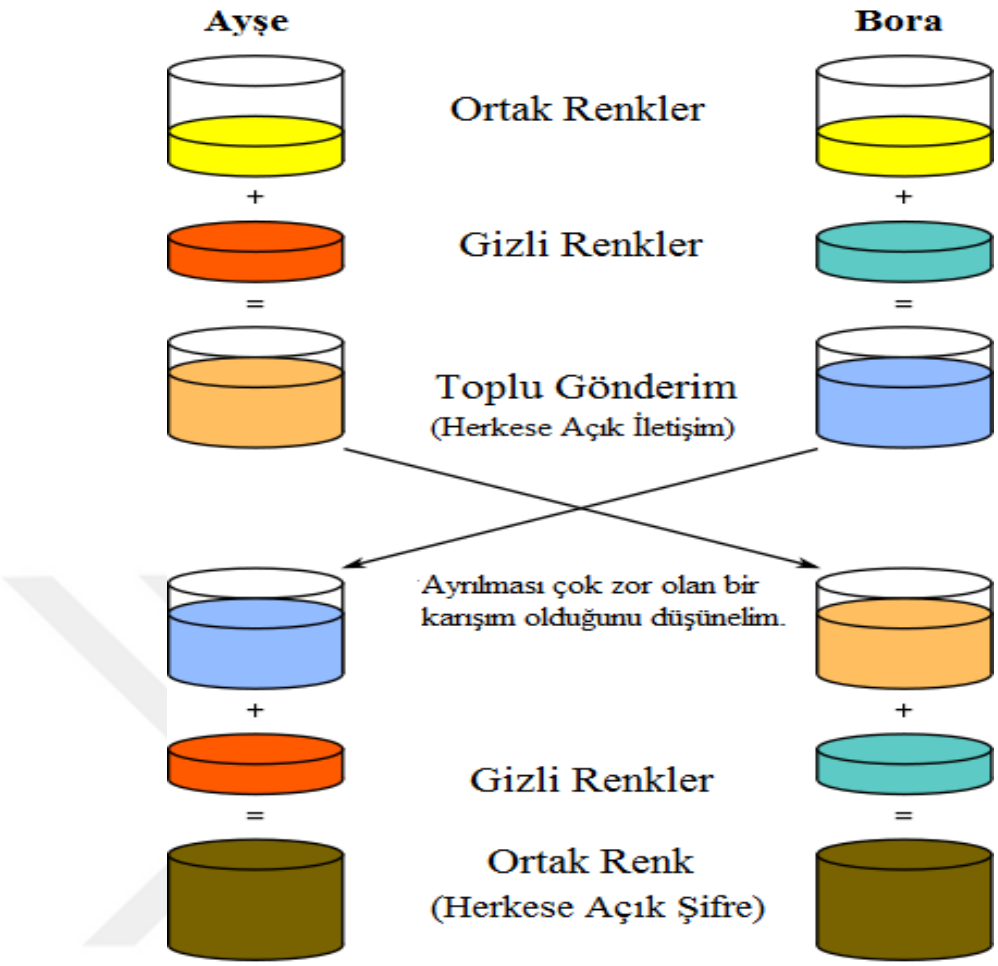
4.2.1. Diffie-Hellman

İlk açık anahtar türü olan Diffie-Hellman, 1976 yılında Diffie-Hellman tarafından "New Directions In Cryptography" isimli makalede yayınlanmıştır. Yayınlandığı günden günümüze kadar olan sürece kadar önemli bir kriptografi türü sayılmıştır ve sayılmaya da devam ediyor. Bir açık anahtar dağıtım şemasıdır.

Sonlu bir cisimde (Galois) ya bir asal sayının modülü veya bir polinomsal cisim üzerinde üstelleştirilmesine dayanır (Tuncal, 2008).

Diffie-Hellman anahtar dağıtım şemasının çalışma yöntemi, güvenli olmayan bir iletişim ağı üzerinden birbirlerini daha öncesinde tanımayan (aksi de mümkün) iki tarafın ortak bir gizli anahtar oluşturmasını amaçlar.

Diffie-Hellman anahtar dağıtım şemasının anlaşılması açısından, önce olayı renk karışımlarıyla açıklamaya çalışalım (Karataş, 2019), sonrasında ise matematiksel olarak izah edelim.



Şekil 4.4. Diffie-Hellman anahtar anlaşması

İlk olarak Ayşe ve Bora bir tane herkesin bileceği ortak renk (sarı) belirlerler. Ardından Ayşe kendi gizli rengini (turuncu), Bora'da kendi gizli rengini (mavi) belirler. Sonrasında herbiri kendi gizli ve açık renklerinin karışımlarından meydana gelen yeni renkleri birbirleri ile değiştirirler. Sonrasında ellerindeki renkler ile tekradan kendi gizli renklerini karıştırırlar. Son durumda her ikisi de aynı renge sahip olmuş olur ve böylece ortak bir anahtar oluşturmuş olurlar.

Şimdi ise matematiksel olarak izah edelim ve bir örnekle pekiştirelim. A ve B kişileri aşağıdaki adımlar sonrasında ortak bir anahtar oluşturabilirler (Yerlikaya, Bulus ve Arda 2005).

- A kişisi, $0 \leq a \leq p-2$ eşitsizliğini sağlayan rastgele bir a sayısını seçer.
- $c = g^a (mod\ p)$' yi hesaplar ve bunu B'ye gönderir.
- B kişisi, $0 \leq b \leq p-2$ eşitsizliğini sağlayan rastgele bir b sayısını seçer.

- $d = g^b (mod\ p)$'yi hesaplar ve bunu A'ya gönderir.
- A, ortak anahtar k'yı şu şekilde hesaplar:

$$k = d^a = (g^b)^a$$

- B, ortak anahtar olan k'yı şu şekilde hesaplar:

$$k = c^b = (g^a)^b$$

Örnek 4.2.1. A ve B kişileri $p = 13$ asal sayısına ve $g = 7$ ilkel (primitif) köküne sahip olsun. A kişisi gizli anahtar olarak $a = 3$ ve B kişisi gizli anahtar olarak $b = 4$'ü seçsin.

- A kişisi $c = g^a\ (mod\ p) \equiv 7^3\ mod\ 13$ 'ü hesaplar ve $c = 5$'i B kişisine gönderir.
- B kişisi ise $d = g^b (mod\ p) \equiv 7^4\ mod\ 13$ ' ü hesaplar ve $c = 9$'u A kişisine gönderir. Son durumda;

A kişisi k ortak anahtarını $(g^b)^a\ mod\ p\ \equiv\ 9^3\ mod\ 13 = 1$,
B kişisi k ortak anahtarını $(g^a)^b\ mod\ p\ \equiv 5^4\ \ mod\ 13 = 1$ olarak bulur.
Böylelikle A ve B kişilerinin ortak anahtarı $k = 1$ bulunmuş olur. $k = 1$ değeri hem A hem de B için bilinen gizli anahtar olmuş olur.

4.2.2. RSA

Asimetrik şifreleme algoritması olan RSA, 1977 yılında R. Rivest, A. Shamir ve L. Adleman adlı kişiler tarafından bulunmuştur. Ardından asimetrik şifreleme algoritmasına uyarlanarak geliştirilmiştir. Bu algoritma türü, asimetrik şifreleme sistemlerinde ve sayısal imza işlemlerinde güvenli bir şekilde kullanılır. En popüler ve pratik algoritma türlerinden biridir.

Mesajları şifrelemek, anahtar değiştirmek ve sayısal imza oluşturmak için kullanılan bir açık anahtarlı sistemdir. RSA, modüler aritmetiği kullanarak üstelleştirmeye dayanan bir açık anahtarlı şifreleme algoritmasıdır (Tuncal, 2008).

RSA algoritması; anahtar üretimi, şifreleme ve şifre çözme olmak üzere 3 kısımdan oluşur.

Anahtar Üretimi: RSA algoritmasında bir ortak ve bir de gizli anahtar bulunur. Ortak anahtar herkesçe bilinir ve mesajları şifrelemek için kullanılır. Gizli anahtar ise şifrelenen mesajı deşifrelemek için kullanılır. RSA anahtarı aşağıdaki adımlara göre oluşturulur.

1. Öncelikle birbirinden farklı rastgele iki tane asal sayı seçilsin. Bunlar p ve q olsun.

2. $n = p.q$ hesaplansın (Burada n açık ve gizli anahtar için mod değeri hesaplanmasında kullanılacaktır).

3. Sonra $\varphi(n) = (p-1).(q-1)$ değeri hesaplanır (Buradaki $''\varphi''$ sembolü totient olarak adlandırılır. Sayılar teorisinde, bir tamsayının o sayıdan daha küçük ve o sayı ile aralarında asal olan sayıların sayısını belirtir).

4. Rastgele bir e tamsayısı seçilsin ve $EBOB(\varphi(n),\ e) = 1$ ve $1 < e <\ \varphi(n)$ şartlarını sağlasın.

5. $e.d \equiv 1\ mod\ \varphi(n)$ için gizli anahtar $''d''$ bulunur. Böylelikle açık anahtar (e,n) ve gizli anahtar (d,n) olmuş olur.

Şifreleme: Bir tane açık metin seçilir (m). A kişisinin açık anahtarı (n,e) ile $c \equiv m^e (mod\ n)$ hesaplanır. Böylece şifreli metin olan c, A şahsına iletilir.

Deşifreleme: Gizli anahtar olan d ile m mesajı çözülür. $m \equiv c^d\ (mod\ n\ )$ değeri hesaplanır.

Bu durum bir örnekle aşağıda izah edilmeye çalışıldı.

Örnek 4.2.2.

- $p = 7$ ve $q = 13$ olan iki tane asal sayı seçilsin.
- $n = p.q = 91$ olur.
- $\varphi(n) = (p-1).(q-1) = 6.12 = 72$ hesaplanır.
- Bir tane e sayısı seçilsin, öyle ki $\varphi(n) = 72$ ve $ebob(\varphi(n),\ e) = 1;\ e < \varphi(n)$, buradan $e = 5$ seçilir.
- Öyle bir d sayısı belirlenir ki; $d.e \equiv 1\ (mod\ 72)$ ve $d < 72$. Dolayısıyla d için doğru değer 29' dur.
- Sebebi ise $29.5 = 145 \equiv 1\ (mod\ 72)$ olmasıdır.
- Böylece anahtarlardan açık anahtar = (5,91), gizli anahtar ise (29,91) olur.

Yukarıdaki örneğin şifrelenmesi için; bir tane açık metin olan rastgele bir $m = 9$ sayısı seçilsin. Burada A kişisinin açık anahtarı (5,91) ile $c \equiv 9^5 (mod\ 91)$ değeri hesaplandığında c değeri 25 olarak hesaplanır.

Deşifrelemesi için; $m \equiv 25^{29} (mod\ 91)$ değeri hesaplandığında da $m = 9$ gizlenmiş metnine ulaşılmış olur.

RSA Algoritmasında Hız

RSA şifreleme algoritmasının güvenilirliğini arttırmak için çok büyük asal sayılar tercih edilir. Şifreleme ve deşifreleme işlemleri bu çok büyük asal sayılar kullanılarak gerçekleştirildiğinden dolayı sonuca ulaşma süresi uzamaktadır. Bundan dolayı çok büyük mesajların RSA ile şifrelenmesi çok tercih edilmez (Montgomery, 1985; Scheinder, 1996; Stallings, 1998).

Şifreleme ve deşifreleme sürelerini azaltmak için çeşitli algoritmalar geliştirilmiştir. Çok büyük asal sayılar kullanılarak işlem yapılan RSA için Montgomery tarafından 1985 yılında Montgomery algoritması oluşturulmuştur. Bu algoritmanın hızını düşüren diğer bir sebep, kullanılan anahtar boyutlarının büyük olmasıdır. Bu duruma çözüm bulmak için 1990 yılında Wiener tarafından Rebalanced CRT-RSA ve 2003 yılında da Paixo ve Alison tarafından RPrime RSA algoritmaları üretilmiştir. Ayrıca RSA algoritmasının şifre çözme hızını takriben 4 kat oranında arttıran CRT-RSA algoritması, Quisquater ve Couvreur tarafından 1982 senesinde sunulmuştur. Ardından Collins vd. (1997), CRT-RSA algoritmasını kullanarak şifre çözme kısmında MultiPrime-RSA algoritmasını sunmuşlardır. (Aslanyürek, Yerlikaya; 2019)

4.2.3. El Gamal

El Gamal, 1985 yılında Diffie-Hellman anahtar dağıtım algoritmasını kullanarak daha farklı bir açık anahtar kriptosistemi önerisini ortaya attı. Diffie-Hellman anahtar dağıtım algoritmasında olduğu gibi El Gamal'ın güvenliği de sonlu cisimlerde ayrık logaritma probleminin çözümünün zorluğuna dayanır.

El Gamal asimetrik şifreleme sistemi; anahtar üretilmesi, şifreleme ve deşifreleme olmak üzere üç aşamadan oluşmaktadır (Kahate 2013).

Anahtar Üretimi

- Büyük bir p asal sayısı seçilir ve $mod\ p$' ye göre çarpımsal grup olan Z_p^* 'i tanımlar. Bu grupta ilkel (primitif) kök olarak $g \in Z_p^*$ sayısı seçilir.
- $1 \leq x \leq p-2$ olacak şekilde rastgele bir $x \in Z$'yi belirler. $g^x\ mod\ p$ değerini hesaplar.
- x değerini kendi gizli anahtarı olarak muhafaza eder ve (p, q, g^x) açık anahtarlarını yayınlar.

Şifreleme

B kişisi, A kişisinin açık anahtarını kullanarak m şifresiz metnini aşağıdaki gibi şifreler.

- m şifresiz metni $\{0,1, \ldots, p-1\}$ aralığında yer alan bir tamsayıdır.
- $1 \leq k \leq p-2$ olmak üzere bir $k \in Z$ belirlenir. Buradaki k değeri geçici anahtar olarak adlandırılabilir.
- $c_1 = g^k\ mod\ p$ ve $c_2 = m.(g^x)^k\ mod\ p$ değerlerini hesaplar.
- $C = (c_1, c_2)$ ise m şifresiz metninin şifrelenmiş hali olarak yayınlanır.

Şifre Çözme Aşaması

- x gizli anahtarı ile ${c_1}^{-x} = g^{-xk}\ mod\ p$ değerini hesaplar.
- ${c_1}^{-x}.c_2$ ile m şifresiz metnine ulaşılır:

$$ {c_1}^{-x}.c_2 \equiv g^{-xk}.m.g^{xk} \equiv m\ mod\ p $$

Örnek 4.2.3.

$p = 23,\ g = 11,\ k = 3$, gizli anahtar $x = 6,$ gönderilmek istenen mesaj $M = 10$ olsun. $y = g^x (mod\ p)$ için değerleri yerine yazıldığında $y = 11^6\ (mod\ 23) = 9$ açık anahtarını elde edilir. Mesaj şifrelenmek istenirse $c_1 = g^k\ (mod\ p)$ ve $c_2 = y^k.M(mod\ p)$ ifadelerine verilenler yerine yazıldığında;

$$ c_1 = 11^3\ (mod\ 23) = 20 $$

$$ c_2 = 9^3.10\ (mod\ 23) = 22 $$

olur. Dolayısıyla şifreli metin $C(20,22)$ olur. Mesajın deşifrelenmesi için ise yine yukarıda verilen formüllerde verilenler yerine yazılırsa tekrar ana mesaja yani orijinal metne ulaşılmış olur.

$$\frac{22}{20^6}(mod\ 23)$$

$$= 20^6(mod\ 23) = 16\ (mod 23)$$

$$\frac{22}{16}(mod\ 23) = 10 = M$$

4.3. Anahtar Dağıtım Şemaları

Bu kısımda anahtar dağıtım şemalarından Needham-Schroeder, Denning-Sacco, Kerberos ve Bellare-Rogaway'dan bahsedilecektir. Bu kısımdan bahsedilirken önemli ölçüde Stinson (2006)' dan yararlanılmıştır.

Aşağıda vereceğimiz protokol ve formüllerde kullanılan kısaltmalar şu şekildedir.

A: Ahmet **B**: Burcu **TA**: Sunucu

$\boldsymbol{K_{Ahmet}}$: Ahmet'in Anahtarı $\boldsymbol{K_{Burcu}}$: Burcu'nun Anahtarı $\boldsymbol{t_{Burcu}}$: Burcu'nun Bileti

$\boldsymbol{t_{Ahmet}}$: Ahmet'in Bileti **K**: Oturum Anahtarı

4.3.1 Needham-Schroeder Anahtar Dağıtım Şeması

İlk oturum anahtarı dağıtım şemalarından biri, 1978 de ortaya atılan Needham-Schroeder anahtar dağıtım şemasıdır. Bir simetrik şifreleme algoritmasıdır. Daha sonra göreceğimiz Kerberos protokolünün altyapısını oluşturur. Bu dağıtım şemasının amacı taraflar arasında daha fazla iletişimi sağlamaktır. Sistemin işleyişi ağdaki taraflar arasındaki karşılıklı kimlik doğrulaması esasına dayanır.

Bu şema aşağıda verilen protokolle açıklanmaya çalışıldı. Bu dağıtım şemasında gerekli olan bazı geçerlilik kontrolleri vardır. Buradaki geçerlilik kontrolünden kasıt; şifresi çözülen verilerin doğru formata sahip olduğunu ve beklenen bilgileri içerdiğini doğrulamayı ifade eder. Bu geçerlilik kontrolleri aşağıdaki gibidir (Burada kişiler Ahmet ve Burcu olsun).

1. Ahmet y_1 şifresini çözdüğünde düz metin olan $d_{K_{\text{Ahmet}}}(y_1)$ 'de bazı K ve t_{Burcu} için ;

$$d_{K_{\text{Ahmet}}}(y_1) = r_A \,||\, ID(\text{Burcu})|| \, K \,||\, t_{\text{Burcu}}$$

biçiminde olup olmadığını kontrol eder. Yukarıdaki koşul geçerliyse, Ahmet ''kabul eder'', aksi takdirde ''reddeder'' ve oturumu iptal eder.

2. Burcu; y_3 şifresini çözdüğünde, düz metin olup olmadığını kontrol eder.

$$d_K(y_3) = r_B - 1$$

Bu koşul geçerliyse Burcu ''kabul eder''; aksi halde ''reddeder''.

Needham-Schroeder Protokolü

1. Ahmet rastgele bir r_A sayısını seçer. Sonra kendi kimliğini ve Burcu'nun kimliğini r_A 'dan TA'ya gönderir.

2. TA rastgele bir K oturum anahtarını seçer ve hesaplar;

$$t_{\text{Burcu}} = e_{K_{\text{Burcu}}}(K \,||\, ID(\text{Ahmet}))$$

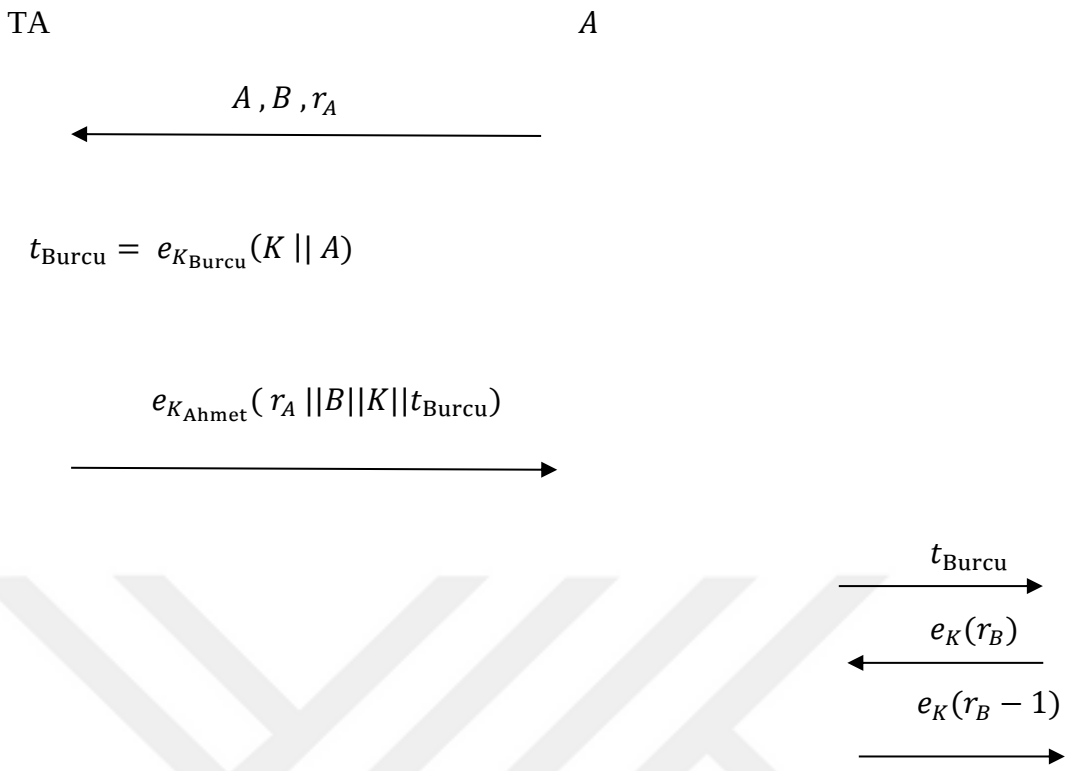
$y_1 = e_{K_{\text{Ahmet}}}(r_A \,||\, ID\ (\text{Burcu})|| \, K \,||\, t_{\text{Burcu}}\ )$ olur ve y_1'i Ahmet'e gönderir.

3. Ahmet, K_{Ahmet} anahtarını kullanarak K ve t_{Burcu}'yu elde etmek için y_1'in şifresini çözer. Sonra, t_{Burcu}'yu Burcu'ya gönderir.

4. Burcu, K'yı elde etmek için K_{Burcu} anahtarını kullanarak t_{Burcu}'nun şifresini çözer. Ardından rastgele bir r_B sayısı seçer ve $y_2 = e_K(r_B)$ değerini hesaplar. Sonra y_2'yi Ahmet'e gönderir.

5. Ahmet, K oturum anahtarını kullanarak y_2'nin şifresini çözer ve r_B 'yi elde eder. Daha sonra $y_3 = e_K(r_B - 1)$ değerini hesaplar ve y_3'ü Burcu'ya gönderir. Böylece süreç tamamlanmış olur.

Needham-Schroeder bilgi akışı aşağıdaki **Şekil 4.5**'te ifade edilmiştir.



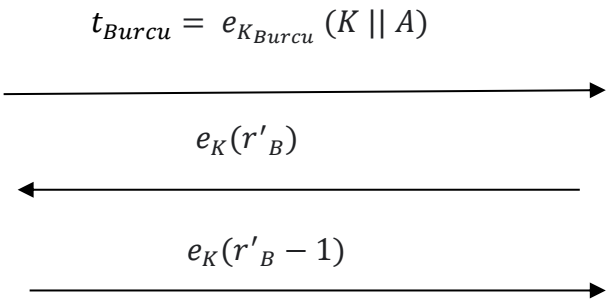
Şekil 4.5. Needham-Schroeder bilgi akışı

4.3.2. Denning-Sacco Anahtar Dağıtım Şeması

Needham-Schroeder protokolü, ortak anahtar sistemlerinde gizli anahtarların değiş tokuş edilmesi usulüne dayanıyordu. Ancak bu durum güvenlik zaafiyetlerine sebebiyet veriyordu. 1981 yılında Denning ve Sacco bu zaafiyetten yola çıkarak kendi adlarını taşıyan yeni bir protokol tasarladılar. Bu protokolün işleyişi şu şekildedir.

Ahmet'in Burcu ile arasındaki Needham-Schroeder anahtar dağıtım şemasının bir oturumunu kaydettiğini (bu oturuma S diyelim) ve bir şekilde bu S oturumu için K oturum anahtarını aldığını varsayalım. Sonra Onur, S oturumunun üçüncü akışından başlayarak Burcu ile Needham-Schroeder anahtar dağıtım şemasının yeni bir oturumunu (örneğin S'), daha önce kullanılmış olan bileti (t_{Burcu}) , Burcu'ya göndererek başlatabilir.

Onur Burcu



Burcu, $e_K(r'_B)$ ile yanıt verdiğinde, Onur; bilinen K anahtarını kullanarak bunun şifresini çözebilir, 1'i çıkarır ve ardından sonucu şifreleyebilir. $e_K(r'_B - 1)$ değeri, S' oturumunun son akışında Burcu'ya gönderilir. Burcu ise bunun şifresini çözer ve kabul eder.

Bu durumun olası sonuçları aşağıda irdelenmeye çalışıldı.

Onur ve Burcu arasındaki S' oturumunun sonunda Burcu, Ahmet ile paylaşılan yeni bir oturum anahtarı olan K'ya sahip olduğunu düşünür. (Bunun sebebi Ahmet'in kimliğinin t_{Burcu} biletinde yer alıyor olmasıdır). Bu K anahtarı Onur tarafından bilinir ancak Ahmet tarafından bilinemeyebilir. Çünkü Ahmet, Burcu ile önceki oturumdan sonra K anahtarını atmış olabilir. Yani S oturumu sonlandırılmıştır. Bu nedenle Burcu'nun bu saldırı tarafından aldatılmasının iki yolu vardır:

1. S' oturumunda dağıtılan K anahtarı, Burcu'nun iletişim kuracağı Ahmet tarafından bilinmiyor olması.

2. S' oturumunun anahtarı olan K, Burcu'nun iletişim kuracağı kişi değil de başka biri tarafından biliniyor olması. (yani Onur tarafından biliniyor)

4.3.3 Kerberos

Kerberos, 1980'lerin sonunda ve 1990'ların başında MIT'de geliştirilen, oturum anahtarı dağıtımı için popüler bir dizi şemadan oluşan protokol türüdür. Kerberos protokolü 3 farklı altyapıdan oluşur. Bunlar;

- Anahtar Dağıtım Merkezi (Key-Distribution-Center/KDC)
- Kullanıcı ve hesabı
- İstenilen servisi sağlayan sunucu (Badur, 2013).

Burada bahsi geçen anahtar dağıtım merkezinin temel olarak 2 farklı görevi vardır.

- Kimlik doğrulama hizmeti (Authentication Service)
- Bilet Sağlama Hizmeti (Ticket-Granting Service)

Kerberos protokolünde bir kullanıcı ilk defa hizmet alacaksa sırasıyla aşağıdaki aşamaları geçmek zorundadır. Bunlar;

- Kimlik doğrulama hizmeti takası
- Bilet sağlama hizmeti takası
- Kullanıcı/Sunucu takası

Needham-Schroeder protokolünde olduğu gibi Kerberos protokolünde de bazı geçerlilik kontrolleri vardır. Bunlar aşağıdaki gibidir:

1. Ahmet y_1 şifresini çözdüğünde, düz metin olan $d_{K_{Ahmet}}(y_{1)}$ 'yı bazı K ve L değerleri için ;

$$d_{K_{Ahmet}}(y_1) = r_A \,||\, ID(Burcu) || \, K \,||\, L$$

biçiminde olup olmadığını kontrol eder. Bu koşul geçerli değil ise, Ahmet ''reddeder'' ve mevcut oturumu iptal eder.

2. Burcu, y_2 ve t_{Burcu} şifresini çözdüğünde, düz metin olan $d_K(y_2)$'nin ;

$$d_K(y_2) = ID(Ahmet) \,||\, time$$

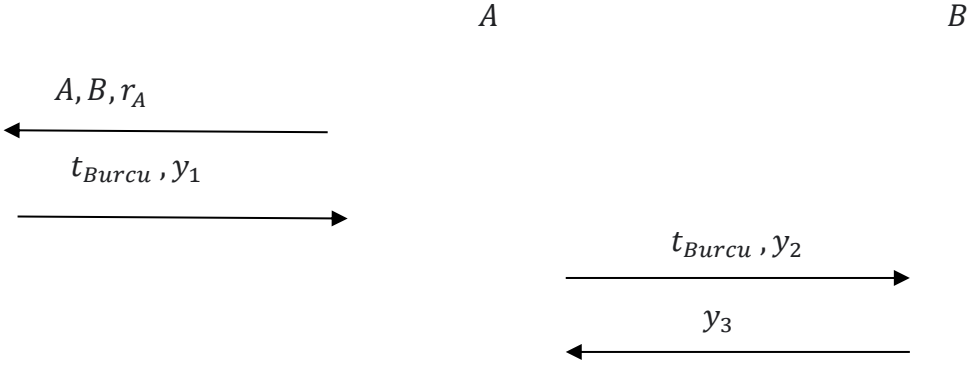
formunda olup olmadığını kontrol eder, ve $d_{K_{Burcu}}(t_{Burcu})$ düz metni için,

$$d_{K_{Burcu}}(t_{Burcu}) = K \,||\, ID(Ahmet) \,||\, L$$

formunda Ahmet'in kimliği hem düz metinlerde hem de $time \leq L$ 'de aynıdır. Bu koşullar geçerliyse, Burcu ''kabul eder''; aksi takdirde ''reddeder''.

3. Ahmet y_3 şifresini çözdüğünde, $d_K(y_3) = time + 1$ olduğunu kontrol eder. Eğer bu koşul geçerliyse, o zaman Ahmet ''kabul eder''; aksi halde ''reddeder''.

Kerberos protokolünün akışı aşağıda **Şekil 4.6'** da gösterilmiştir.



Şekil 4.6. Kerberos'un Bilgi Akışı

$$A = ID(Ahmet)$$
$$B = ID(Burcu)$$
$$t_{Burcu} = e_{K_{Burcu}}(K \,||\, A \,||\, L\,),$$
$$y_1 = e_{K_{Ahmet}}(r_A \,||\, B \,||\, K \,||\, L)$$
$$y_2 = e_K(\,A \,||\; time), \text{ ve}$$
$$y_3 = e_K(time + 1\,).$$

Kerberos protokolünün işleyişi aşağıdaki gibi olmaktadır.

1. Ahmet, rastgele bir r_A sayısı seçsin. Kendi kimliğini (ID), Burcu'nun kimliğini (ID) ve r_A 'yı TA' ya göndersin.

2. TA rastgele bir K oturum anahtarını ve bir geçerlilik süresi olan L'yi seçer. Sonra Burcu'nun biletini hesaplar,

$$t_{Burcu} = e_{K_{Burcu}}(K \,||\, ID(Ahmet) \,||\, L)$$

ve

$$y_1 = e_{K_{Ahmet}}(r_A \,||\, ID(Burcu)||\text{K}||\text{L})$$

Ardından TA , t_{Burcu} ve y_1'i Ahmet'e gönderir.

3. Ahmet , K_{Ahmet} anahtarını kullanarak y_1'in şifresini çözer ve K'yı yakalar. Ardından şimdiki zamanı yani $time$'ı belirler ve aşağıdaki gibi hesaplar.

$$y_2 = e_K(ID(Ahmet) \,||\; time\,).$$

Sonra Ahmet, t_{Burcu}'yu ve y_2'yi Burcu'ya gönderir.

4. Burcu, K_{Burcu} anahtarını kullanarak t_{Burcu}'un şifresini çözer ve K'yı elde eder. Ayrıca K anahtarını kullanarak $y_2{}'$ nin şifresini çözer ve $time$'ı elde eder. Sonra aşağıdaki denklemi hesaplar.

$$y_3 = e_K(time + 1\,)\,.$$

En sonda da Burcu, Ahmet'e y_3'ü gönderir. Ve böylece akış tamamlanmış olur.

Kerberos protokolü özetle analiz edilecek olursa, Ahmet kişisinden TA'ya herhangi bir oturum için istek gönderildiğinde TA, rastgele bir K isminde oturum anahtarı üretir. Ayrıca yine TA , K'nın geçerli olacağı $lifetime$'ı yani L'yi belirler. Burada, K oturum anahtarı , L zamanına kadar geçerli bir anahtar olarak kabul edilir. Tüm bu veriler Ahmet'e gönderilmeden önce şifreleme işlemine tabi tutulur. Ahmet ,

y_1 şifresini çözmek için gizli anahtarını kullanabilir ve böylece K ve L' yi elde edebilir. Geçerli zamanın, anahtarın kullanım ömrü içinde olduğunu ve y_1'in, Ahmet'in rastgele meydan okumasını (r_A'yı içerdiğini) doğrulayacaktır. Ayrıca y_1'in, Burcu'nun kimliğini içerdiğini doğrulayabilir. Burada Burcu, Ahmet'in iletişim kurmak için hedeflediği kişidir. Yapılan bu kontroller, Onur'un bir önceki oturumda TA tarafından iletilmiş olabilecek 'eski' bir y_1'i tekrar oynatmasını engeller. Ahmet, t_{Burcu}'yu, Burcu'ya aktarır. Ayrıca Ahmet, yeni oturum anahtarı K'yı, mevcut zaman $time$'ı ve Ahmet'in ID'sini şifrelemek için kullanacaktır. Sonra elde edilen şifreli metin olan y_2'yi Burcu'ya gönderir. Burcu, Ahmet'ten t_{Burcu} ve y_2'yi aldığında K, L ve Ahmet'in ID'sini elde etmek için t_{Burcu}'nun şifresini çözer. Sonra y_2'nin şifresini çözmek için yeni oturum anahtarı K 'yı kullanır ve t_{Burcu} ve y_2'den şifresi çözülen Ahmet'nin ID'sinin aynı olduğunu doğrular.

Bu durum, Burcu'ya t_{Burcu} içinde şifrelenen oturum anahtarının y_2'yi şifrelemek için kullanılan anahtarla aynı olmasını sağlar. Ayrıca, K anahtarının süresinin dolmadığını doğrulamak için $time \leq L$ 'yi de kontrol etmesi gerekir. Son olarak, Burcu yeni oturum anahtarı K'yı kullanarak $time + 1$ 'i şifreler ve sonucu Ahmet'e geri gönderir. Ahmet bu mesajı aldığında y_3'ü kullanarak K 'nın şifresini çözer ve sonucun $time + 1$ olduğunu doğrular. Ahmet'in oturumunun y_3 mesajını üretmek için K gerekli olduğundan , K anahtarı Burcu'ya başarıyla iletilmesi gereklidir. L kullanım ömrünün amacı, Needham-Schroeder anahtar dağıtımına yapılan Denning-Sacco saldırısında yapıldığı gibi aktif bir düşmanın ''eski'' mesajları daha sonra yeniden iletmek üzere saklamasını önlemektir.

Kerberos protokolünün dezavantajlarından biri, belirli bir K oturum anahtarının geçerli olup olmadığını belirlemek için geçerli saat kullanıldığından, ağdaki tüm kullanıcıların senkronize (eş zamanlı) saatlere sahip olması gerektiğidir. Uygulamada, mükemmel senkronizasyon sağlamak çok zordur, bu nedenle bir miktar zaman farklılıklarına izin verilmelidir.

Kerberos ve Denning-Sacco Kıyası

1. Kerberos protokolünde karşılıklı anahtar doğrulaması gerçekleştirilir. Ahmet, kendi ID 'sini şifrelemek için yeni oturum anahtarı K'yı kullanarak Burcu'yu, K'nın değerini bildiğine ikna etmeye çalışıyor. Benzer şekilde Burcu $time + 1$'i, K'yı kullanarak şifrelediğinde, Ahmet'in K'nin değerini bildiğini gösteriyor.

2. Needham-Schroeder'de Burcu'ya yönelik bilgiler iki kez şifrelenir: Halihazırda şifrelenmiş olan Burcu'nun bileti, Ahmet'in gizli anahtarı kullanılarak yeniden şifrelenir. Bu durum şemaya gereksiz karmaşıklık kattığı için kerberos'ta bu çift şifreleme işlemi kaldırıldı.

3. Denning-Sacco saldırısına karşı kısmi koruma, Kerberos'ta geçerli zamanın(yani, genellikle zaman damgası olarak adlandırılan $time$ değerinin) yaşam süresi L içinde olduğunu doğrulayarak sağlanır. Temel olarak bu, zaman aralığını sınırlar.

Needham-Schroeder ve Kerberos protokolleri, günümüzün anahtar dağıtım şemalarında genellikle kullanışlı olarak kabul edilmeyen bazı özelliklere sahiptir. Bunlara kısaca bakalım.

1. Zaman dalgaları güvenilir, eş zamanlı(senkronize) saatler gerektirir. Zaman dalgalarını kullanan şemaları analiz etmek zordur ve onlar için ikna edici güvenlik kanıtları vermek zordur. Bu nedenle, mümkünse genellikle zaman damgaları yerine rastgele sorgulamaların kullanılması tercih edilir.

2. Anahtar onayı, bir oturum anahtarı dağıtım şemasında mutlaka olması gereken bir özellik değildir. Bu nedenle, anahtar dağıtım şemalarından anahtar onayının çıkarılması artık sıklıkla tavsiye edilmektedir.

3. Needham-Schroeder ve Kerberos'ta hem gizlilik hem de özgünlük sağlamak için şifreleme kullanılır. Ancak, gizlilik için şifreleme ve özgünlük sağlamak için bir mesaj doğrulama kodu kullanılması tercih edilir. Örneğin, Needham-Schroeder'da çift şifrelemeyi kaldırabilir ve kimlik doğrulama için MAC'ları aşağıdaki gibi kullanabiliriz.

TA rastgele bir K oturum anahtarını seçer. Ardından;

$$y_1 = (e_{K_{Burcu}}(K), MAC_{Burcu}(ID(Ahmet) \,||\, e_{K_{Burcu}}(K))),$$

ve

$$y_1{}' = e_{K_{Ahmet}}(K), MAC_{Ahmet}\,(ID(Burcu)\ ||r_A||\ e_{K_{Ahmet}}(K)))$$

ifadelerini hesaplar. TA, y_1 ve $y_1{}'$ yi Ahmet'e gönderir. O'da daha sonra y_1 i Burcu'ya iletir.

4. Denning-Sacco saldırısını önlemek için şemanın akış yapısı değiştirilmelidir. Denning-Sacco tipi tekrar saldırılarını önlemek için, herhangi bir "güvenli" şema, Burcu'yu oturum anahtarını almadan önce aktif bir katılımcı olarak içermelidir.

Çözüm; TA'ya oturum anahtarı için bir istek göndermeden önce Ahmet'in Burcu ile iletişim kurmasını (veya tam tersi) gerektirir.

4.3.4. Bellare-Rogaway Şeması

Bellare ve Rogaway tarafından 1995 yılında önerilmiştir. Aşağıda bu şemanın protokolünün nasıl işlediğinden bahsedilmiştir. Bu protokol, diğer protokollerden farklı bir akış yapısına sahiptir. Ahmet ve Burcu, TA'ya gönderilen rastgele görevleri seçerler. Böylece Burcu, TA oturum anahtarını vermeden önce plana dahil olur.

TA'nın Ahmet'e gönderdiği bilgiler şunlardan oluşur:

1. Bir oturum anahtarı(Ahmet'in gizli anahtarı kullanılarak şifrelenir),

2. Şifreli oturum anahtarı için bir MAC, Ahmet ve Burcu'nun kimlikleri ve Ahmet'in meydan okuması.

Burcu'ya gönderilen bilgiler de benzerdir.

Bellare-Rogaway anahtar dağıtım şeması prokolünün işleyişi aşağıdaki gibidir.

1. Ahmet, rastgele bir r_A sayısı seçer ve kendi ID'sini, Burcu'nun ID'sini ve r_A 'yı Burcu'ya gönderir.

2. Burcu, rastgele bir r_B sayısı seçer ve kendi ID'sini, Ahmet'in ID'sini, r_A ve r_B'yi, TA'ya gönderir.

3. TA, rastgele bir K oturum anahtarını seçer. Sonra;

$$y_B = (e_{K_{Bob}}(K), MAC_{Bob}(ID(Alice)|| ID(Bob) || r_B || e_{K_{Bob}}(K)))$$

ve

$$y_A = (e_{K_{Alice}}(K), MAC_{Alice}(ID(Bob)|| ID(Alice) || r_A || e_{K_{Alice}}(K)))$$

değerleri hesaplanır.

TA, y_B'yi Burcu'ya ve y_A'yı Ahmet'e gönderir. Ve akış tamamlanmış olur.

Ahmet ve Burcu, ilgili MAC'leri geçerliyse "kabul eder". Örneğin Burcu, şifreli oturum anahtarları olan $y_{B,1}$ ve $y_{B,2}$ 'yi aldığında, bunu aşağıdaki gibi doğrular.

$$y_{B,2} = MAC_{Burcu}(ID(Ahmet) || ID(Burcu)|| r_B || y_{B,1}$$

Bu şemada herhangi bir anahtar onayı verilmez. Örneğin, Ahmet kabul ettiğinde, Burcu'nun kabul edip etmediğini veya Burcu'nun TA tarafından gönderilen mesajı alıp almadığını bile bilmez. Ahmet kabul ettiğinde, bu sadece beklediği bilgiyi aldığı ve bu bilginin geçerli olduğu (veya daha doğrusu MAC

geçerli olduğu) anlamına gelir. Ayrıca, bu oturum anahtarı, Ahmet'in gizli anahtarı kullanılarak şifrelendiğinden; Ahmet, az önce aldığı bilgilerden başka hiç kimsenin, K oturum anahtarını hesaplayamayacağından emindir. Oturum, Burcu'nun bakış açısından incelendiğinde analiz benzer olacaktır. Başka bir deyişle, anahtar doğrulama (tek yönlü veya çift yönlü) hedefi anahtar dağıtım şemasından kaldırılmış oldu. Bunun yerine, şemadaki ''kabul eden'' bir katılımcının bakış açısından, iletişime geçilecek asıl kişilerin dışında hiç kimsenin yeni oturum anahtarını hesaplayamaması gibi, biraz daha zayıf (ama yine de yararlı) hedef alınır.

Bir düşmanın (iletişimdeki art niyetli kişi) amacı; iletişim kuracak asıl kişilerin haricindeki kişilerin, oturum anahtarı olan K değerini bilmesini sağlamaktır. Örneğin, iletişimde asıl kişi Ahmet ve iletişim kurmayı amaçladığı kişi Burcu olsun. Düşman Onur veya başka bir ağ kullanıcısı, oturum anahtarını hesaplayabilirse amacına ulaşır. Öte yandan, oturum anahtarını hesaplayabilen tek ağ kullanıcısı Ahmet ise, Onur'un saldırısı başarılı sayılmaz. Bu durumda Burcu, oturum anahtarını hesaplayamaz ancak başka kimse de (Ahmet dışında) hesaplayamaz.

Yukarıdaki ifadelerden hareketle güvenli bir oturum anahtarı dağıtım şemasını tanımlamak için aşağıdaki maddelere ihtiyaç duyulur:

1. İletişimi sağlayacak olan Ahmet, Burcu ve TA'nın dürüst olduğu,

2. Şemada kullanılan şifreleme şeması ve MAC'nin güvenli olduğu

3. Gizli anahtarların yalnızca amaçlanan sahipleri tarafından bilindiği

4. Mükemmel rastgele sayı üreteçleri kullanılarak rastgele zorlukların oluşturulduğu varsayımları yer alır.

5. Son olarak, TA'nın mükemmel bir rastgele sayı üreteci kullanarak oturum anahtarları oluşturduğu varsayılır.

Onur'un (iletişimde art niyetli 3. Kişi) saldırı gerçekleştirebileceği çeşitli yolları ele alalım. Bu olasılıkların her biri için küçük bir ihtimal dışında gerçekleşme olasılığı oldukça düşüktür.

1. Onur, pasif bir düşmandır.

2. Onur, aktif bir rakip ve Ahmet'te bu planın meşru bir katılımcısı. Onur; Burcu veya TA'nın kimliğine bürünebilir ve plan sırasında gönderilen mesajları engelleyebilir ve değiştirebilir.

3. Onur, aktif bir rakip ve Burcu'da bu planın meşru bir katılımcısı. Onur, Ahmet'i veya TA'yı taklit edebilir ve şema sırasında gönderilen mesajları engelleyebilir ve değiştirebilir.

Yukarıda sayılan olası saldırıları aşağıdaki gibi analiz edebiliriz.

1. Düşman pasif ise, o zaman Ahmet ve Burcu, katılımcı oldukları herhangi bir oturumda ''kabul et'' çıktısını verecektir. Ayrıca, her ikisi de aynı oturum anahtarının (K) şifresini çözebilecektir. Şifreleme şeması güvenli olduğundan, başka hiç kimse (Onur dahil) K değerini hesaplayamaz.

2. Ahmet, oturumun bir katılımcısı olsun. Sadece Burcu'nun ve kendisinin bilebileceği yeni bir oturum anahtarı almak istesin. Ancak Burcu ile gerçekten iletişim kurup kuramayacağını bilemez. Çünkü Onur, Burcu'yu taklit ediyor olabilir.

Ahmet y_A mesajını aldığında, MAC'ın geçerli olup olmadığını kontrol eder. Bu etiket, Ahmet'in rastgele meydan okuması r_A'nın yanı sıra Ahmet ve Burcu'nun kimliklerini ve şifreli oturum anahtarı $e_{K_{Ahmet}}(K)$ 'yı içerir. Bu durum Ahmet'i, etiketin TA tarafından yeni hesaplandığına ikna eder, çünkü TA, Ahmet dışında MAC_{Ahmet} anahtarını bilen tek taraftır. Ayrıca, rastgele meydan okuma r_A, önceki bir oturumdan etiketin tekrar oynatılmasını engeller.

Son olarak, etikete $e_{K_{Ahmet}}(K)$ eklenmesi, bir rakibin TA tarafından seçilen oturum anahtarını başka bir şeyle değiştirmesini önler. Onur (iletişimde art niyetli kişi), şemanın geçerli oturumunda Burcu'nun kimliğine bürünmüş olsa bile, Ahmet K oturum anahtarının şifresini çözebilen diğer tek kullanıcı olan Burcu'dan emin olabilir.

3. Burcu'nun katılımcı olduğu herhangi bir oturumda, normalde yalnızca Ahmet'in ve kendisinin bilebileceği yeni bir oturum anahtarı elde etmeleri gerekir. Ancak Burcu, Ahmet ile gerçekten iletişim kurup kuramadığından tam emin olamaz. Çünkü Onur, Ahmet'i taklit ediyor olabilir.

Burcu, y_B mesajını aldığında, etiketin geçerli olup olmadığını kontrol eder. Bu etiket, Burcu'nun rastgele sorgulamasını, r_B'yi ayrıca Ahmet ve Burcu'un kimliklerini ve şifreli oturum anahtarı $e_{K_{Burcu}}(K)$'yı içerir. Bu, Burcu'yu MAC'ın TA tarafından yeni hesaplandığına ikna eder. Çünkü TA, Burcu dışında MAC_{Burcu} anahtarını bilen tek taraftır. Son olarak MAC'a, $e_{K_{Burcu}}(K)$'un dahil edilmesi, bir rakibin TA tarafından seçilen oturum anahtarını başka bir şeyle değiştirmesini önler.

Bu nedenle Burcu, K oturum anahtarının şifresini çözebilen diğer kullanıcının Ahmet olduğuna ikna olur.

5. SONUÇ VE ÖNERİLER

Haberleşme ve iletişim, geçmişten günümüze değin insanlar ve devletler arasındaki ilişkilerde önemli bir etken olarak görülmüştür. İnsanlar karşısındaki kişilerle iletişimini çeşitli yol ve yöntemlerle gerçekleştirmişlerdir. Bu yol ve yöntemler teknolojinin de gelişmesiyle günümüzde çok farklı bir hal almıştır. Özellikle 3. kişilerin (art niyetli) iletişime dahil olmasını bertaraf etmek amacıyla iletişimde çeşitli şifreleme ve deşifreleme sistemleri kullanılmıştır. Bu şifreleme sistemleri yalnızca iletişimde değil artık teknolojinin de gelişmesiyle çok farklı alanlarda da kullanılmaya başlamıştır. Devletler arası haberleşme, internet üzerinden yapılan alışverişler vs. bunların başında gelmektedir. Eskiden olduğu gibi günümüzde de bu tür faaliyetleri engelleyici ya da zarar verici etkenlerden muhafaza etmek için güvenli şifreleme işlemine ihtiyaç duyulmaktadır. Bu şifreleme işlemi, gün geçtikçe artan bu zararlı etkenlere karşı kendisini daha da geliştirmeye ve kırılması (çözülmesi) daha güç bir yapıya bürünmesi ihtiyacını doğurmuştur. Gelişen teknoloji ile birlikte zararlı yazılımlar sebebiyle kullanılan şifreleme sistemlerinin güvenliği tehdit altındaydı. Bu durum şifrelemede daha kompleks algoritmaların kullanımını gerektirdi.

Şifreleme teknikleri kendi içerisinde güçlü ve zayıf yönler barındırmaktadır. Bu yüzden kullanacağımız şifreleme algoritması hakkında detaylı bilgi sahibi olmamız gerekir. Şifreleme algoritmaları, kullanım amacına göre farklılık göstermektedir. Tezimizde de bahsedildiği üzere küçük boyutlu bir şifreleme için en ideal algoritma blowfish, daha büyük boyutlu algoritmalar için ise DES ve AES kullanımı tavsiye edilmektedir. Blowfish çalışma süresi bakımından en verimli algoritma türlerindendir. Güvenlik ön planda tutulacak ise AES'in kullanımı, kullanılacak hafıza bakımından ise DES'in kullanımı tavsiye edilmektedir.

Yine bu tez çalışması ile geçmişten günümüze kadar olan süreçte kullanılan şifreleme yöntemleri incelendi ve örneklerle izah edilmeye çalışıldı. Şifrelemeyle alâkalı günümüzde kullanılan en yaygın ve kullanışlı şifreleme türleri hakkında bilgi verildi. Kriptolojinin en son geldiği noktada kullanılan anahtar dağıtım şemaları detaylı olarak incelendi ve birbiri ile karşılaştırıldı. Ayrıca kriptoloji biliminin "Matematik"le nasıl sıkı ilişki içerisinde olduğu anlatıldı.

Bu alanla alakalı yapılan Türkçe kaynaklı çalışmalar az ve sınırlı olduğu için yaptığımız bu çalışmanın, bundan sonraki çalışmalara ışık tutacağı kanaatindeyiz.

KAYNAKLAR

Aghayev, M. (2017). *Kriptoloji ve veri şifreleme teknikleri üzerine*. (Yayımlanmamış yüksek lisans tezi). Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir.

Altındiş, H. (1999). Sayılar Teorisi ve Uygulamaları. Kayseri: Erciyes Üniversitesi Yayın Komisyonu.

Yerlikaya, T., & Aslanyürek, C. (2019). RSA algoritmasının şifreleme hızını arttıran algoritmalar ve performansları. *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi, 10*(3), 853-862.

Aydoğan, M. (2014). *Adli bilişimde görüntü üzerine kriptografi uygulamaları*. (Yayımlanmamış yüksek lisans tezi). Fırat Üniversitesi, Elazığ.

Babaoğlu, A., (2009). ‘’Kriptolojinin Geçmişi: Bir Şifreleme Algoritması Kullanmadan Önce Son Kullanım Tarihine Bakın!’’, 500, Bilim Teknik, Temmuz.

Bahçetepe, H. (2006). *Modüler çarpma algoritmalarının incelenmesi ve kriptolojide uygulamaları*. (Yayımlanmamış yüksek lisans tezi). İstanbul Üniversitesi, İstanbul.

Başkök, M. (2007). *AES şifreleme algoritmasının modellenmesi*. (Yayımlanmamış yüksek lisans tezi). Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara.

Balcı, M. (1985). Matematik Analiz 1. Ankara Üniversitesi Fen Fakültesi Yayınları.

Badur, B. (2013). Kerberos protokolü nedir? İşleyişi nasıldır? https://forum.mshowto.org

Buluş, H.N.(2006). ‘’Temel Şifreleme Algoritmaları ve Kriptanalizlerin İncelenmesi’. Yüksek Lisans Tezi,Trakya Üniversitesi, Edirne.

Çallıalp, F. : Soyut Cebir ve Sayılar Teorisi, Ondokuzmayıs Üniv. Fen-Ed. Fak. Yay. No:12, 1986.

Çallıalp, F. (2009). ‘’Sayıların Teorisi’’. Birsen Yayınevi.

Çimen, C. (2007). ‘’Şifrelerin matematiği Kriptografi’’. Yüksek Lisans Tezi, ODTÜ, Ankara.

Denton, B. (2011). ‘’Evulation of Cryptographic Construction Properties and Security Requirements of Modern Secure Hashing Algorithms’’. Master Thesis, Alabama University, Alabama, USA.

Diffie W.; Hellman ,M. E. , (1976): “New Directions in Cryptography ” , IEEE Transactions on Information Theory, IT-22, No.6, 644-654.

Ersin, G. (2006). ‘’TÜBİTAK UEKAE Açık Anahtar Altyapısı Eğitim Kitabı’’ Mayıs (2006).

Frovol, M. (2021). Şifreleme Nedit?. https://elfanet.com.tr/tr/main/article/sifreleme-nedir/26 adresinden 19.07.2022 tarihinde erişilmiştir.

Gilbert, L., Gilbert, J. (2009). Elements of Modern Algebra.

Hosseınpour, S. (2018). ‘’Matematiksel İfadelerin Üretimi ve Çözümüne Dayalı Bir Kriptoloji Yöntemin Tasarımı ve Gerçeklenmesi’’. Doktora Tezi, Karadeniz Teknik Üniversitesi , Fen Bilimleri Enstitüsü, Trabzon.

İmamoğlu, K. (2011). ‘’Görüntü sıkıştırma ve internet üzerinden güvenli aktarım

Sistemi’’. Yüksek Lisans Tezi, Kocaeli Universitesi, Fen Bilimleri Enstitüsü, Kocaeli.

http://www.bilmuh.gyte.edu.tr/~ispinar/BM550/SECURITYCRS9-13.pdf, (09.02.2004)

Kahate, A. (2013). *Cryptography and network security*. Tata McGraw-Hill Education.

Koblitz, N. (1994). *A course in number theory and cryptography* (Vol. 114). Springer Science & Business Media.

Kocabıyık, Ö (2021). ‘’Anahtar anlaşma protokollerinin incelenmesi üzerine’’. Yüksek Lisans Tezi , Ağrı İbrahim Çeçen Üniversitesi.

Külen, F. (2013). ‘’Kriptolojik bazı şifreleme yöntemlerinde cebirsel yaklaşımlar’’. Yüksek Lisans Tezi, Gaziosmanpaşa Üniversitesi. Tokat.

Menezes, A.J., Oorschot, P.C. Van ve Vanstone, S.A., (1997). Handbook of Applied

Cryptography Chapter. 11, CRC Press, inc., USA.

Montgomery, P. L. (1985). Modular multiplication without trial division. *Mathematics of computation*, *44*(170), 519-521.

Özyılmaz, Ç. (2014). ‘’Kriptolojiye Giriş’’. Yüksek Lisans Tezi, Karabük Üniversitesi, Fen Bilimleri Enstitüsü, Karabük.

Sağıroğlu, Ş. , Alkan, M. , ‘’Her Yönüyle Elektronik İmza’’, Grafiker Yayınları, 1st ed., Ankara, 8-9, 21-50 (2005).

Schneider B. (1996). Applied Cryptography.New York.

Senay, H. (2007). Sayılar teorisi dersleri. Dizgi Ofset Matbaacılık, Konya.

Soyalıç, S. ,(2004) “Kriptografik Hash fonksiyonları ve uygulamaları”, Yüksek Lisans Tezi, Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Kayseri, 1-3, 15-18, 24-26.

Stallings W. (1998). Cryptography and Network Security: Principles and Practice. Prentice Hall. ISBN 0-13-869017-0.

Stinson, D.R., (1995). Cryptography Theory and Practice, CRC Press LLC.

Stinson, D. R., “Classical Cryptograph, Cryptography Theory and Practice”, Ed:

Rosen, K. H., Chapman & Hall / CRC, New York, 2: 1- 20 (2002).

Stinson, D.R., (2006). ‘’Cryptography, Theory and Practice’’, 3nd Edition. Kenneth H. Rosen(Ed.), s.414-421.

Şiap, V., (2008). Matris Kodlar ile Mceliece Şifreleme Sistemi. (Yüksek Lisans Tezi), Matematik Bölümü, Sakarya Üniversitesi, Sakarya.

Tuncal, T.,(2008). “Bilgisayar güvenliği üzerine bir araştırma ve şifreleme-deşifreleme üzerine uygulama”, Yüksek Lisans Tezi, Maltepe Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Weis, S.(2007). ‘’Theory and Practice of Cryptography’’, Google Tech Talks.

http://www.youtube.com/watch?v=IzVCrSrZIX8 (23.12.2013).

Yerlikaya, T. , Bulus, E. , Arda, D. (2005)., “Asimetrik Kriptosistemler ve Uygulamaları”, II. Mühendislik Bilimleri Genç Arastırmacılar Kongresi, İstanbul, 24-31 .