T.C.

TRAKYA ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

HOMOMORFİK ŞİFRELEME YÖNTEMİ ÜZERİNE BİR İNCELEME

RAŞİT ERKAN HOŞCOŞKUN

YÜKSEK LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİMDALI

Tez Danışmanı: Prof. Dr. M. Tolga SAKALLI

EDİRNE-2020

RAŞİT ERKAN HOŞCOŞKUN'un hazırladığı "**HOMOMORFİK ŞİFRELEME YÖNTEMİ ÜZERİNE BİR İNCELEME**" başlıklı bu tez, tarafımızca okunmuş, kapsam ve niteliği açısından Bilgisayar Mühendisliği Anabilim Dalında bir **Yüksek lisans tezi** olarak kabul edilmiştir.

Jüri Üyeleri (Unvan, Ad, Soyad):	İmza
Prof. Dr. M. Tolga SAKALLI	.....
Dr. Öğr. Üyesi Andaç ŞAHİN MESUT	.....
Dr. Öğr. Üyesi Bora ASLAN	.....

Tez Savunma Tarihi: 15/01/2020

Bu tezin Yüksek Lisans tezi olarak gerekli şartları sağladığını onaylarım.

Prof. Dr. M. Tolga SAKALLI

.....

Tez Danışmanı

Trakya Üniversitesi Fen Bilimleri Enstitüsü onayı

.....

Prof. Dr. Murat YURTCAN

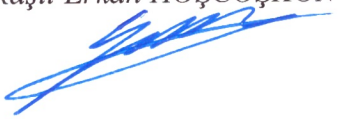
Fen Bilimleri Enstitüsü Müdürü

T.Ü. FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİMDALI YÜKSEK LİSANS PROGRAMI DOĞRULUK BEYANI

Trakya Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında, tüm verilerin bilimsel ve akademik kurallar çerçevesinde elde edildiğini, kullanılan verilerde tahrifat yapılmadığını, tezin akademik ve etik kurallara uygun olarak yazıldığını, kullanılan tüm literatür bilgilerinin bilimsel normlara uygun bir şekilde kaynak gösterilerek ilgili tezde yer aldığını ve bu tezin tamamı ya da herhangi bir bölümünün daha önceden Trakya Üniversitesi ya da farklı bir üniversitede tez çalışması olarak sunulmadığını beyan ederim.

15/01/2020

Raşit Erkan HOŞCOŞKUN



Yüksek Lisans Tezi

Homomorfik Şifreleme Yöntemi Üzerine Bir İnceleme

T.Ü. Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

ÖZET

Bu tez kapsamında Homomorfik Şifreleme Yönteminin ve çeşitli uygulama alanlarının örneklerinin incelenmesi yapılacaktır. Bu kapsamda, Kısmi-Homomorfik Şifrelemenin temellerini oluşturan RSA Algoritması (Rivest, Shamir & Adleman, 1978), ElGamal Algoritması (ElGamal, 1984) ve Paillier Algoritması (Paillier, 1999) ve Tamamen Homomorfik Şifreleme (Fully Homomorphic Encryption) Algoritmaları detaylı incelenecektir. Tezin, bilgi güvenliği ve kriptoloji alanına bilgi anlamında katkı sağlaması öngörülmektedir.

Yıl : 2020

Sayfa Sayısı : 56

Anahtar Kelimeler : Homomorfik Şifreleme, Tamamen Homomorfik Şifreleme, Kısmi-Homomorfik Şifreleme, Homomorfik Şifreleme uygulama alanları

Master's Thesis

An Investigation on Homomorphic Encryption Method

Trakya University Institute of Natural Sciences

Departmen of Computer Engineering

ABSTRACT

In this thesis, Homomorphic Encryption Methods and various application areas are studied. In this context, RSA Algorithm (Rivest vd., 1978) which forms the basis of Partial-Homomorphic Encryption, ElGamal Algorithm (ElGamal, 1984) and Paillier Algorithm (Paillier, 1999) and Fully Homomorphic Encryption Algorithms are examined in detail. It is expected that this thesis will contribute theoretically to the fields of information security and cryptology.

Year : 2020

Number of Pages : 56

Keywords : Homomorphic Encryption, Fully Homomorphic Encryption, Partial Homomorphic Encryption, Homomorphic Encryption applications

TEŞEKKÜR

Tüm akademik yaşamım boyunca bilgisi, tecrübesi ve dostluğuyla bana daima destek olan danışmanım Sayın Prof. Dr. M. Tolga SAKALLI'ya (Trakya Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü Bölüm Başkanı),

Çalışmaya yaptığı matematiksel katkıları için, Sayın Prof. Dr. Figen ÖKE'ye (Trakya Üniversitesi Fen Fakültesi Matematik Bölümü Bölüm Başkanı),

Çalışmamın her aşamasında yardımlarını esirgemeyen bölüm hocalarıma ve arkadaşlarıma,

Bugünlere gelmemi sağlayan ve hep yanımda olan aileme,

Yol arkadaşım, sevgili eşim Ayşenur'a,

Sonsuz teşekkürler…

İÇİNDEKİLER

ÖZET iv
ABSTRACT v
TEŞEKKÜR vi
İÇİNDEKİLER vii
SİMGELER VE KISALTMALAR x
ÇİZELGELER ve ŞEKİLLER xi
BÖLÜM 1 1
GİRİŞ 1
1.1. Kriptoloji 1
1.2. Kriptografinin Gelişimi 3
1.3. Modern Kriptoloji 5
1.3.1. Simetrik Şifreleme 5
1.3.2. Asimetrik Şifreleme 9
1.4. Tezin Önemi 10
BÖLÜM 2 12
MATEMATİKSEL ALT YAPI 12
BÖLÜM 3 16
KULLANILAN KRİPTOLOJİK ALGORİTMALAR 16
3.1. Giriş 16
3.2. Diffie-Hellman anahtar değişimi 16
3.3. RSA Algoritması 17
3.3.1. Algoritmanın Altyapısı 19
3.3.2. RSA Algoritmasına Karşı Saldırılar 21
3.4. ElGamal Algoritması 23
3.4.1. Algoritmanın Altyapısı 23
3.4.2. ElGamal Algoritmasına Karşı Saldırılar 25
3.5. Paillier Algoritması 25
3.5.1. Algoritmanın Altyapısı 25
BÖLÜM 4 28
HOMOMORFİK ŞİFRELEME 28

4.1. Giriş **28**
4.2. Homomorfik Şifreleme **29**
4.3. Kısmi-Homomorfik Şifreleme **31**
4.3.1. RSA Algoritması ve HE **31**
4.3.2. ElGamal Algoritması ve HE **31**
4.3.3. Paillier Algoritması ve HE **32**
4.3. Homomorfik Benzeri Şifreleme Sistemleri **33**
4.3.1. Boneh-Goh-Nissim (BGN) Şifreleme **34**
4.4. Tam-Homomorfik Şifreleme **35**
4.4.1. İdeal Kafes Tabanlı FHE **37**
4.4.2. Tam Sayı Tabanlı FHE **38**
4.4.3. Hata ile Öğrenme (LWE) Tabanlı FHE **39**
4.4.4. NTRU Benzeri FHE **41**
BÖLÜM 5 ..... **43**
HOMOMORFİK ŞİFRELEME UYGULAMALARI **43**
5.1. Giriş **43**
5.2. Genom ve HE **43**
5.3. Ulusal Güvenlik ve HE **44**
5.3. Eğitim ve HE **44**
5.4. Sağlık ve HE **45**
5.5. Bulut Bilişim ve HE **45**
5.6. Mobil Ajan ve HE **46**
5.7. Oylama ve HE **46**
5.8. Avuç ve Parmak İzi ve HE **46**
5.9. Finans ve HE **46**
5.10. Reklam ve HE **47**
5.11. Kablosuz Algılayıcı Ağlarda Güvenli Veri Kümeleme ve HE **47**
5.12. Internet of Things (IoT) ve HE **48**
5.13. Veri Tabanı Güvenliği ve HE **48**
5.14. Ağ Kontrol Sistemleri ve HE **49**
5.15. Kapalı Teklif Açık Arttırma ve HE **49**
BÖLÜM 6 ..... **51**
SONUÇ **51**
KAYNAKLAR **53**

ÖZGEÇMİŞ.....56

SİMGELER VE KISALTMALAR

AES	: Advenced Encryption Standard
DES	: Data Encryption Algorithm
ECC	: Elliptic Curve Cryptosystem
FHE	: Fully Homomorphic Encryption
HE	: Homomorphic Encryption
IDEA	: International Data Encryption Algorithm
IoT	: Internet of Things
NIST	: National Institute of Standars and Technology
PGP	: Pretty Good Privacy
SHA	: Secure Hash Algorithm
SWHE	: Somewhat Homomorphic Encryption

ÇİZELGELER ve ŞEKİLLER

Şekil 1.1. Şifreleme işlemi 2
Şekil 1.2. Şifre çözme işlemi 3
Şekil 1.3. Simetrik şifreleme işlemi 5
Şekil 1.4. Simetrik şifre çözme işlemi 6
Şekil 1.5. Simetrik şifreleme anahtar sayısı hesaplama 7
Şekil 1.6. Akış şifreleri 8
Şekil 1.7. Blok şifreler 8
Şekil 1.8. Asimetrik şifreleme işlemi 10
Şekil 1.9. Asimetrik şifre çözme işlemi 10
Şekil 3.1. Diffie-Hellman anahtar değişimi 17
Şekil 3.2. James H. Ellis şifreleme fikri 18
Şekil 3.3. James H. Ellis şifre çözme fikri 18
Şekil 3.4. RSA algoritması 20
Şekil 3.5. ElGamal algoritması 24
Şekil 3.6. Paillier algoritması 26
Şekil 4.1. Örnek gösterimi 30

BÖLÜM 1

GİRİŞ

İnsanlık küçücük bir noktadan gelişimine başlamıştır. Bu gelişim sırasında insanoğlu değişik ihtiyaçlara gerek duymuştur. Bu ihtiyaçlardan en önemlilerinden bir tanesi bilgi gizliliğidir.

En gizli bilgi insanın kendi zihninde sakladığı bilgidir. Bu bilgiyi insan ikinci bir şahıs ile paylaşmak istemediği sürece gizliliğini sürekli koruyacak, paylaşmak istediğinde ise gizliliğini yitirme riski doğacaktır. Tamda bu noktada kriptoloji bilimi devreye girerek hem bu bilginin paylaşılmasını hem de bu bilginin gizli kalmasını sağlamaktadır.

Günümüzde hemen hemen bütün bilgilerimizin, gerçekleştirdiğimiz işlemlerin elektronik ortamda olmasından ve iletişimimizi büyük bir bölümünü elektronik ortamda gerçekleştirdiğimizden dolayı kriptolojiye olan ihtiyacımız giderek artmaktadır.

1.1. Kriptoloji

Kriptoloji, özünde bir matematik bilimidir (Tuncay, 2015, s. 1-2). Paylaşılmak istenilen bir bilginin kötü ve istenmeyen kişiler tarafından ele geçirilmesi durumunda anlaşılamayacak bir hale getirilmesi için kullanılan bir yöntemdir (Sakallı, 2006, s. 2). Kriptoloji, kriptografi ve kriptanaliz olarak iki alt bilime ayrılmaktadır (Tuncay, 2015, s. 1).

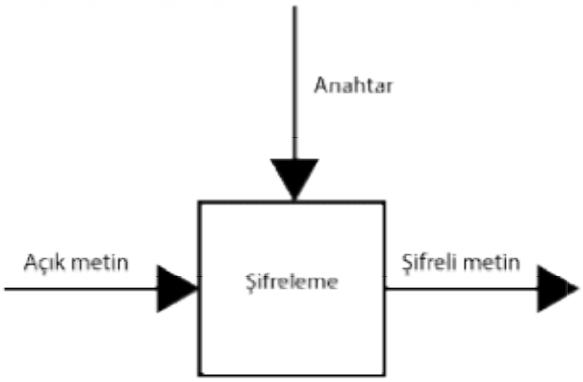
Kriptografi, bilgi güvenliği adı altında 3 temel güvenlik öğesini; bilgi paylaşımı sırasında bilgi kötü ve istenmeyen kişiler tarafından ele geçirilmesi durumuda anlaşılamamasını (veri gizliliği), iletilmek istenen tarafa tam ve eksiksiz olarak ulaştırılmasını (veri bütünlüğü) ve yetkili kişiler tarafından istenildiği zaman ulaşılabilir

ve kullanılabilir durumda olmasını (verinin erişilebilirliği) sağlayan matematiksel yöntemleri içermektedir (Canberk & Sağıroğlu, 2006).

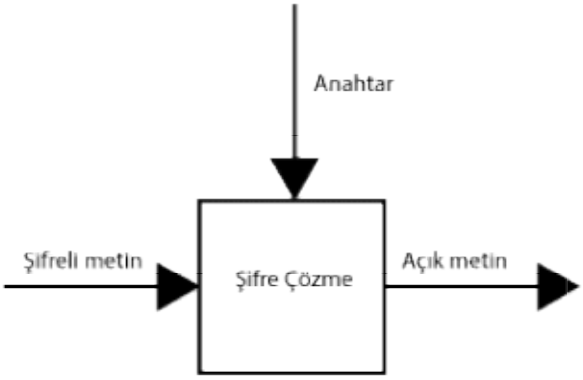
Kriptografi bilgi güvenliğini sağlamak için bazı yapı taşlarına sahiptir (Sakallı, 2006, s. 2);

- Açık metin: Bilginin herhangi bir işlem görmemiş ilk halidir.
- Şifreli metin: Açık metnin belirli matematiksel işlem görmüş halidir.
- Şifreleme: Açık metni şifreli metin haline getirirken üzerinde yapılan matematiksel işlemler bütünüdür.
- Şifre çözme: Şifreli metni açık metin haline getirirken üzerinde yapılan matematiksel işlemler bütünüdür.
- Anahtar: Şifreleme ve şifre çözme işlemlerini gerçekleştirirken kullanılan bir veri parçasıdır.

Şifreleme işlemi Şekil 1.1'de ve şifre çözme işlemi Şekil 1.2'de gösterilmektedir. Şifreleme işlemi gönderen tarafından, şifre çözme işlemi alıcı tarafından yapılmaktadır (Yerlikaya, Buluş & Buluş, 2006, s. 1).



Şekil 1.1. Şifreleme işlemi



Şekil 1.2. Şifre çözme işlemi

Kriptanaliz, kötü veya istenmeyen kişiymiş gibi davranarak, oluşturulan kriptografik yapıya saldırarak, anahtarı bilmeden şifreli metinden açık metni veya anahtarı bulmaya çalışan bir bilim dalıdır. Kriptanaliz, oluşturulan kriptografik yapının güvenliği test ederek geliştirilmesinde büyük bir rol oynadığı için oldukça önemlidir (Tuncay, 2015, s. 2). Kriptanaliz saldırılarının en yaygın dört modeli vardır (Rijmen, Stinson & Biryukov'dan aktaran Sakallı, 2006, s. 9).

- **Sadece Şifreli Metin Saldırısı (Ciphertext-Only):** Bu saldırı modelinde saldırgan sadece şifreli metne sahiptir.
- **Bilinen Açık Metin Saldırısı (Know-Plaintext Attack):** Saldırgan bazı açık metinlere ve bu açık metinlerin şifrelenmiş metin hallerine sahiptir.
- **Seçilmiş Açık Metin Saldırısı (Chosen-Plaintext Attack):** Bilinen açık metin saldırısının aynısıdır. Fakat açık metin ve şifreli metin seçilmiştir.
- **Seçilmiş Şifreli Metin Saldırısı (Chosen-Ciphertext Attack):** Saldırganın saldırdığı kişinin bilgisayarına erişimi olduğu zaman gerçekleşir. Sahip olduğu şifreli metinleri şifre çözme işlemi yapmak için saldırdığı kişinin bilgisayarını kullanır.

1.2. Kriptografinin Gelişimi

Kriptolojinin ilk örneği bundan 4000 yıl önce olduğu tahmin edilen Mısırlıların yazdığı kitabelerdir. Spartalılar tarafından Skytale adı verilen MÖ 450 yıllarında bulunduğu tahmin edilen şerit şeklindeki deri ya da parşömenin belirli boyuttaki bir tahta silindire sarılmasıyla okunabilen yöntem kullanılmıştır. MÖ 3. yüzyılda Kautilya

tarafından yazılmış olan Artha-sastra kriptanaliz yöntemleri içermektedir. MÖ 100-50 yıllarında JuliaCeasar 'Sezar şifrelemesi' olarak bilinen alfabedeki harflerin 3 harf ötelenmesi ile oluşturulan kriptografi yöntemini komutanları ile haberleşmede kullanmıştır. Sezar şifrelemesinin değişik sürümleri uzun süreler kullanılmıştır. 16. yüzyılda Blaise de Vigenere 'Vigenere' şifrelemesi olarak bilinen kriptografik yöntemi buldu ve kriptografi konulu bir kitabında anahtar yapısına yer verdi. Vigenere şifreleme yöntemi Sezar şifreleme yöntemine benzerlik göstermektedir. Sezar şifresinden farkı şifreleme için bir anahtar kullanır ve bu anahtardaki harfin alfabedeki yeri kadar öteleme yapılır (Sakallı, 2006, s. 6). 1854 yılında Charles Wheatstone playfair şifreleme yöntemini buldu. Playfair şifreleme yönteminin anahtarı 5x5 boyutlarında bir matris şeklindedir (Çakır, 2016, s. 103). 20. yüzyılın başlarında Joseph Mauborgne ve Gilbert Vernam kusursuz şifreleme yöntemi olan 'one time pad yöntemini bulmuşlardır. Bu yöntemde anahtar rastgele bir sayı dizisinden oluşturulmaktadır ve bundan dolayı çözülmesi imkânsıza yakındır. 1. ve 2. Dünya Savaşları sırasında yüzlerce kriptografi yöntemi bulunmuştur. 1970 yılında Horst Feistel Data Encryption Standard (DES) algoritmasının ilk adımları olan Lucifer algoritmasını geliştirdi. Lucifer algoritması 128 bit uzunluğunda bir anahtara sahipti. Devamında National Institute of Standars and Technology (NIST) tarafından yapılan geliştirmeler sonucunda anahtar 56 bit uzunluğuna indirildi ve DES ismini alarak duyuruldu. 1977 yılında Ronald L. Rivest, Adi Shamir ve Leonard M. Adleman soyadlarının baş harflerini isim olarak verdikleri RSA algoritmasını geliştirdiler. 1985 yılında Neal Koblitz ve Victor S. Miller Elliptic Curve Cryptosystem (ECC) algoritmasını geliştirdiler. 1990 yıllında Xuejia Lai ve James Massey tarafından International Data Encryption Algorithm (IDEA) bulundu. Phil Zimmerman 1991 yılında Pretty Good Privacy (PGP) bir e-mail ve dosya şifreleme protokolünü geliştirdi. 1995 yılında NIST tarafından Secure Hash Algorithm1 (SHA) hash (özetleme) fonksiyonlarına dayalı bir standart olarak yayınlanmıştır. NIST 1997 yılında DES algoritmasının yerini alabilecek yeni bir simetrik algoritma bulunması için yarışma açtı ve 2001 yılında yarışmayı Joan Daemen ve Vincet Rijmen'nin geliştirmiş olduğu Rijndael algoritması kazandı. NIST tarafından Advenced Encryption Standard (AES) ismini alarak bir standart olarak yerleşti (Ellison, 2011).

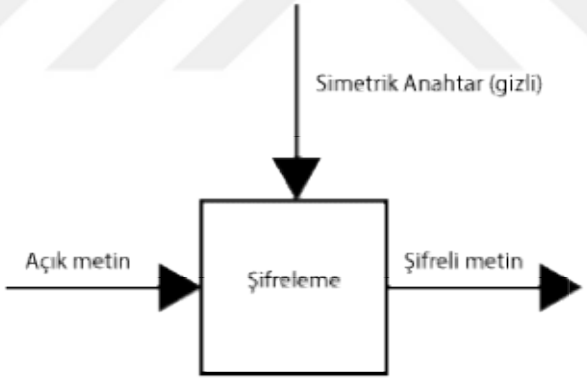
1.3. Modern Kriptoloji

Bölüm 1.2'de anlatılan kriptolojinin gelişiminde DES algoritmasına kadar olan kısım 'İlkel Şifreleme Yöntemleri' olarak adlandırılmaktadır. DES algoritması ile simetrik anahtarların kriptolojiye girmesi ve teknolojinin ilerlemesi ile birlikte kriptoloji yeni bir yönde ilerlemeye başladı (Sakallı, 2006, s. 4).

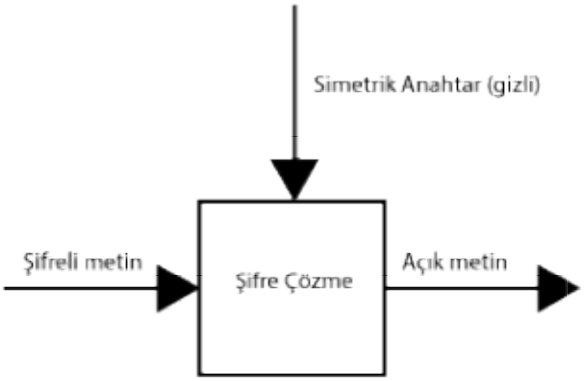
Günümüzde kriptoloji 2 ana başlık altında ayrılmaktadır. Bunlar; simetrik ve asimetrik şifrelemedir.

1.3.1. Simetrik Şifreleme

Simetrik şifrelemede şifreleme işlemi ve şifre çözme işlemleri için aynı anahtarı kullanılmaktadır. Bu yöntemde anahtar muhakkak gizli tutulması gerekmektedir. Eğer anahtar kötü kişilerin eline geçerse yapmış olduğunuz şifreleme işlemini rahatça çözebilecek ve bilgilerinize ulaşabilecektir (Yerlikaya vd., 2006, s. 1). Şekil 1.3 ve Şekil 1.4'de simetrik anahtar ile şifreleme ve şifre çözme işlemleri gösterilmiştir.



Şekil 1.3. Simetrik şifreleme işlemi



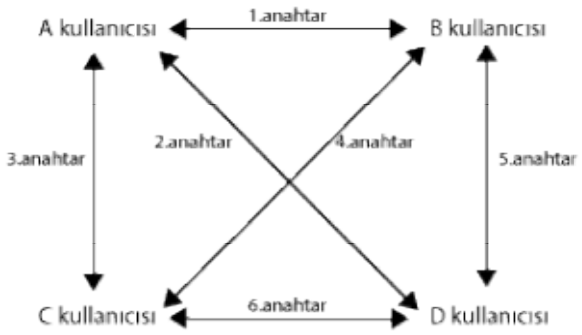
Şekil 1.4. Simetrik şifre çözme işlemi

Simetrik şifrelemenin avantajları (Kodaz & Botsalı, 2010, s. 12);

- Şifreleme ve şifre çözme işlemleri hızlıdır.
- Anahtar boyutu küçüktür.
- Kullanımı anlaşılır ve kolaydır.

Simetrik şifrelemenin dezavantajları (Kodaz & Botsalı, 2010, s. 12);

- Anahtarın güvenliği zordur.
- Anahtarın hem şifreleme tarafında hem de şifre çözme tarafından bulunması gerektiğinden dolayı, anahtarın karşı tarafa ulaştırılması gerekir ve bundan dolayı güvenlik riskleri içerir.
- Şekil 1.5'de görüldüğü gibi her bir iki kullanıcı arasında ayrı ve özel bir anahtar olması gerektiği için çok fazla kullanıcının olduğu ağlarda çok fazla sayıda anahtara ihtiyaç vardır, ölçeklenebilir değildir. n sayıda kullanıcı içeren bir sistemde toplam anahtar sayısı $\frac{n(n-1)}{2}$'dir.



Şekil 1.5. Simetrik şifreleme anahtar sayısı hesaplama

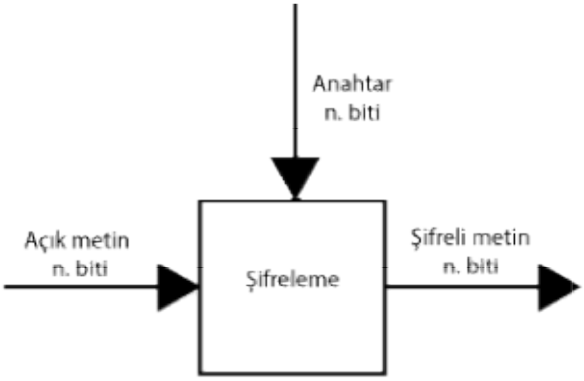
- Kimlik doğrulama gerçekleştiremez. Anahtara sahip olan kötü niyetli biri işlemleri gerçekleştirebilir.
- Veri bütünlüğü sağlayamaz. Anahtara sahip olan kötü niyetli biri bilgileri değiştirmiş olabilir.
- Kimlik doğrulama sağlayamadığı için inkâr edilemez değildir.

DES ve AES simetrik şifreleme algoritmalarına örnek olarak verilebilir (Sakallı, 2006, s. 3).

Simetrik şifreleme kendi içinde akış şifreleri ve blok şifreler olarak iki bölüme ayrılır (Sakallı, 2006, s. 3).

1.3.1.1. Akış Şifreleri

Akış şifreleri açık metin uzunluğunda anahtar ile şifreleme işlemi gerçekleştirir. Açık metindeki her bir karaktere veya bite anahtar üzerinde bir karşılık vardır. Yani açık metindeki her bir karakter veya bit kendine ait bir anahtar ile şifrelenir. Şifreleme işlemini mod 2 de toplamı ya da XOR işlemi sonucu elde edilir. Şekil 1.6'da görüldüğü gibi açık metin n. biti ile anahtar n. biti şifreleme işlemine girerek şifreli metnin n. bitini oluşturulur (Sakallı, 2006, s. 6).

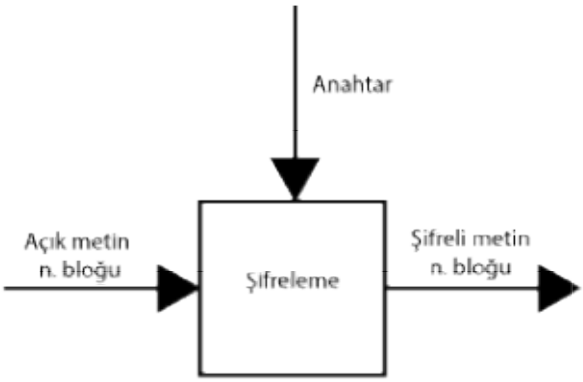


Şekil 1.6. Akış şifreleri

Akış şifrelerinde anahtar rastgele seçildiği ve sadece bir kere kullanıldığı zaman sistem kırılması imkânsız hale gelmektedir. Fakat anahtar boyutunun açık metin ile aynı uzunlukta olması sistemin en büyük dezavantajıdır. One Time Pad ve RC4 algoritmaları akış şifrelerine örnek olarak verilebilir (Sakallı, 2006, s. 3).

1.3.1.2. Blok Şifreler

Blok şifreleri açık metni belirli bir uzunlukta böler ve bloklar halinde şifreleme işlemi gerçekleştirir. Her blok için ayrı bir anahtar yoktur. Tek bir anahtar ile bütün bloklar şifrelenir. Blok şifrelerde şifrelenmiş bloklar kendisinden önce gelen bloklarla ilişkili olabilir veya ilişkili olmayabilir (Sakallı, 2006, s. 4-5).



Şekil 1.7. Blok şifreler

1.3.2. Asimetrik Şifreleme

1970 yıllara kadar kriptografi simetrik anahtar yöntemine dayalı çalışmaktaydı. Asimetrik şifreleme simetrik şifrelemenin dezavantajlarını gidermek için ortaya çıkmıştır. Asimetrik şifrelemede şifreleme ve şifre çözme için ayrı anahtarlar kullanılmaktadır. Şifreleme işleminde kullanılan anahtar açık, şifre çözme işleminde kullanılan anahtar gizlidir. Bu şifrelemede herkesin kendine ait bir açık anahtarı ve bir gizli anahtarı bulunmaktadır. Bilgi göndermek istenilen kişinin açık olan anahtarı ile şifreleme işlemi gerçekleştirilir ve gönderilir. Alıcı kişi gizli anahtarı ile şifre çözme işlemini gerçekleştirir. Bu işlemin tam tersi de geçerlidir. Gizli anahtarı ile şifrelenen bir bilgi bu gizli anahtarın karşılığı olan açık anahtar ile şifre çözülebilir. Yani farklı iki anahtar olmasına karşın bir anahtarın şifrelediğini diğeri şifre çözebilmektedir. Şekil 1.8 ve Şekil 1.9'de asimetrik anahtar ile şifreleme ve şifre çözme işlemleri gösterilmiştir (Yerlikaya vd., 2006).

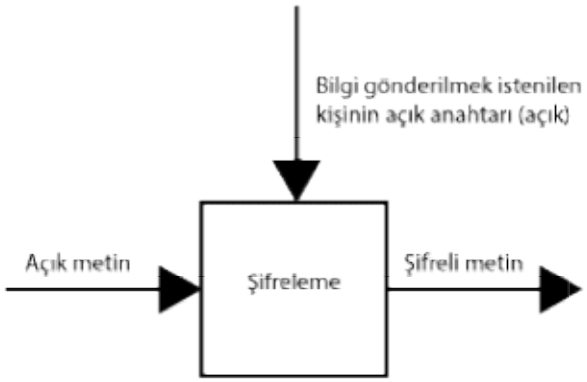
Asimetrik şifrelemenin avantajları (Kodaz & Botsalı, 2010, s. 14);

- Simetrik şifrelemenin aksine herkes eşsiz bir anahtar çiftine sahip olduğu için kimlik doğrulama sağlanır.
- Anahtar fazlalığı yoktur.
- Simetrik şifrelemeye göre daha güvenilirdir.

Asimetrik şifrelemenin dezavantajları (Kodaz & Botsalı, 2010, s. 14);

- Simetrik şifreleme sistemine göre çok daha yavaştır.
- Daha fazla işlem gücü gerektirir.

RSA, ElGamal ve Paillier asimetrik şifreleme algoritmalarına örnek olarak verilebilir (Sakallı, 2006, s. 3).



Şekil 1.8. Asimetrik şifreleme işlemi



Şekil 1.9. Asimetrik şifre çözme işlemi

1.4. Tezin Önemi

Kriptoloji, şifreleme ve şifre çözme işlemlerini gerçekleştirerek bilginin güvenliğini sağlamayı amaçlamıştır. Şifreleme işlemi ile bilginin gizliliğini, şifre çözmeişlemi ile bilginin bütünlüğünü ve anahtar ile bilginin erişilebilirliğini sağlamaktadır.

Şifreleme ve şifre çözme işlemlerinin yanında bilginin gizliliğini sürdürerek bilgi üzerinde işlem yapılabilmesinin gerektiği yerler doğmuştur. Bu ihtiyacı gidermek için Homomorfik Şifreleme (HE) yöntemi ilk olarak 1977 yılında Rivest, Adleman ve Derouzous tarafından gündeme getirilmiştir. Bu tez kapsamında Homomorfik Şifreleme yönteminin ve çeşitli uygulama alanlarının örneklerinin incelenmesi yapılacaktır. Bu kapsamda, Kısmi-Homomorfik Şifreleme yönteminin temellerini oluşturan RSA

Algoritması (Rivest vd., 1978), ElGamal Algoritması (ElGamal, 1984) ve Paillier Algoritması (Paillier, 1999) ve Tam-Homomorfik Şifreleme (FHE) yöntemi incelenecektir.

BÖLÜM 2

MATEMATİKSEL ALT YAPI

Bu bölümde, tez çalışmasında yararlanılan matematiksel ifadelerin anlaşılabilmesi için temel matematiksel alt yapı verilecektir. Verilan tanım, teorem ve önermelerin ispatlarına (Akkaş vd., 1998) (Çelik vd., 2010) (Asar, Arıkan & Arıkan, 2012) (Demir, 2014) (Acar vd., 2018)'den ulaşılabilir.

Tanım 2.1: $a, b \in Z^+$olsun. $a > 1$ olsun. $1 \leq b < a$ aralığındaki tüm b tam sayıları için, $\frac{a}{b} \in Z^+$ değerleri C kümesinin elemanı olsun. $s(C) = 1$ ise a sayısına asal sayı denir (Çelik vd., 2010).

Tanım 2.2: $P(x)$ polinomu, sabit olmayan ve derecesi $P(x)$ in derecesinden küçük olan polinomların çarpımı olarak yazılabiliyorsa bu polinomlardan her birine $P(x)$ polinomunun bir çarpanıdır denir (Çelik vd., 2010).

Tanım 2.3: a bir reel sayı ve b pozitif bir tamsayı olmak üzere,

$$a^b = \underbrace{a \times a \times a \ldots \times a}_{b \text{ tane}}$$

biçimide tanımlanan sayıya a'nın b-inci kuvveti denir ve a'ya taban, b'ye üs (kuvvet) adı verilir (Çelik vd., 2010).

Tanım 2.4: $a \in Z^+$ olsun. $x, y \in Z$ için, $x - y$ sayısı a sayısına bölünüyorsa x sayısı y sayısına a modülüne göre denktir, denir ve

$x \equiv y \ (mod\ a)$ veya $x\ mod\ a \equiv y$

yazılır (Çelik vd., 2010).

Teorem 2.1: a, b ve c $\in Z^+$ olsun. a $\equiv$ b $(mod\ c)$ ise b $\equiv$ a $(mod\ c)$'dir (Çelik vd., 2010).

Teorem 2.2: a, b, c ve d $\in Z^+$ olsun. $a^d \equiv b\ (mod\ c)$ ise b $\equiv a^d (mod\ c)$'dir (Çelik vd., 2010).

Teorem 2.3: a, b, c ve d $\in Z^+$ olsun. $a^d \equiv 1\ (mod\ c)$ ise $a^{d\times b} \equiv 1\ (mod\ c)$ ve $a^{d+1} \equiv 1\ (mod\ c)$'dir (Çelik vd., 2010).

Tanım 2.5: a $\in Z^+$ olsun. a ile aralarında asal ve a'dan küçük pozitif tam sayıların sayısı $\varphi(a)$ olduğuna göre, Z^+'dan Z'ye tanımlanan φ fonksiyonuna *Euler phi fonksiyonu* denir (Akkaş vd., 1998).

Teorem 2.4: a $\in Z^+$olsun. a tam sayısı asal ise,

$$\varphi(a) = (a - 1)$$

dir (Akkaş vd., 1998).

Teorem 2.5: a, b $\in Z^+$ olsun. a ve b sayıları asal sayı ise,

$$\varphi(a \times b) = (a - 1) \times (b - 1)$$

dir (Akkaş vd., 1998).

Tanım 2.6: G boş olmayan bir küme ve G üzerinde bir * ikili işlemi tanımlı olsun. Eğer

(i) $*$ işlemi birleşme özelliğini sağlarsa; yani, her $a, b, c \in G$ için

$(a * b) * c = a * (b * c)$ ise;

(ii) her $a \in G$ için

$$a * e = e * a = a$$

olacak biçimde bir $e \in G$ varsa (e ye G nin birim elemanı denir);

(iii) her $a \in G$ için

$$a * a' = a' * a = e$$

olacak biçimde bir $a' \in G$ varsa (a' ne a nın bir ters elemanı denir)

o zaman $(G, *)$ sıralı ikilisine bir **grup** denir

Eğer $(G, *)$ grubunda fazaladan her $a, b \in G$ için

$$a * b = b * a$$

ise bu gruba **değişmeli** ya da **abelyan** denir (Demir, 2014).

Tanım 2.7: G bir grup ve A, G nin boş olmayan bir altkümesi olsun. Eğer her $a, b \in A$ için $ab \in A$ ise o zaman A, G nin işlemine göre **kapalı** denir (Demir, 2014).

Tanım 2.8: G bir grup ve H, G nin boş olmayan bir altkümesi olsun. Eğer H, G nin işlemine göre kapalı ve bu işleme göre bir grup ise o zaman H ya G nin **altgrubu** denir ve $H \leq G$ ya da $G \geq H$ ile gösterilir (Demir, 2014).

Tanım 2.9: G bir grup ve $a_1, \ldots, a_n \in G$ olsun. Eğer $G = \langle a_1, \ldots, a_n \rangle$ ise o zaman $a_1, \ldots, a_n$ elemanlarına G'nin **üreteçleri** denir (Asar vd, 2012).

Tanım 2.10: R, boş olmayan bir küme olsun. R üzerinde her $a, b \in R$ için

$+: (a, b) \to a + b$, $\times: (a, b) \to a \times b$

biçiminde tanımlı ve sırasıyla, toplama ve çarpma denilen "+" ve "×" ikili işlemleri verilsin.

Eğer

(i) $(R, +)$ bir abelyan grup ise,

(ii) çarpma birleşme özelliğini sağlarsa; yani, her $a, b, c \in R$ için

$(a \times b) \times c = a \times (b \times c)$ ise,

(iii) R üzerinde dağılma özellikleri sağlanırsa; yani, her $a, b, c \in R$ için

$a \times (b + c) = a \times b + a \times c$ (sol dağılma özelliği)

$(a + b) \times c = a \times c + b \times c$ (sağ dağılma özelliği) ise,

o zaman $(R, +, \times)$ sıralı üçlüsüne bir **halka** denir (Demir, 2014).

Tanım 2.11: R ve S iki halka ve $\varphi: R \to S$ fonksiyonu verilsin. Eğer her $a, b \in R$ için

$\varphi(a + b) = \varphi(a) + \varphi(b)$ ve $\varphi(a \times b) = \varphi(a) \times \varphi(b)$

ise φ ye R den S ye bir **halka homomorfizması** denir (Demir, 2014).

Tanım 2.12: L boştan farklı bir küme ve " $\leq$ " L üzerinde bir bağıntı olsun. L'ye sıralı küme denir. $\Leftrightarrow$

(i) $\forall\, a \in L$ için $a \leq a$

(ii) $\forall\, a, b \in L$ için $a \leq b$ ve $b \leq a$ ise $a = b$

(iii) $\forall\, a, b, c \in L$ için $a \leq b$ ve $b \leq c$ ise $a \leq c$

L sıralı kümesi $(L, \leq)$ notasyonu ile gösterilir (Demir, 2014).

Tanım 2.13: $(L, \leq)$ bir sıralı küme olsun. L'ye **kafes** denir $\Leftrightarrow \forall\, a, b \in L$ için $Sup\{a, b\} = a \vee b$ve $Inf\{a, b\} = a \wedge b$ mevcuttur (Acar vd., 2018).

Tanım 2.14: R bir halka ve I, R nin bir toplamsal altgrubu olsun. Eğer her $a \in I$ ve $r \in R$ ise $ra \in I$ ise I ya R nir bir **sol ideali** ve her $a \in I$ ve $r \in R$ için $ar \in I$ ise I ya R nin bir **sağ ideali** denir. Eğer I hem sol ideal ve hem de sağ ideal ise I ya R nin bir **ideali** denir (Acar vd., 2018).

Tanım 2.15: Kafes, bağımsız vektörlerin $b_1, b_2, \ldots, b_n$'nin lineer kombinasyonlarıdır. Bir kafes formülü;

$$L = \sum_{i=1}^{n} \overrightarrow{b_i} \times v_i\,, v_i \in Z$$

şeklinde tanımlanır ve her bir vektör $b_1, b_2, \ldots, b_i$, L **kafesinin temeldir.** Temel vektörler hemen hemen ortogonal ise temele **'iyi'** denir, aksi takdirde kafesin **'kötü'** temeli olarak adlandırılır (Acar vd., 2018).

BÖLÜM 3

KULLANILAN KRİPTOLOJİK ALGORİTMALAR

3.1. Giriş

Tezin bu bölümünde Homomorfik Şifreleme yönteminin temellerini oluşturan RSA algoritması, ElGamal ve Paillier algoritmaları ve bu algoritmaların bazılarının temelini oluşturan Diffie-Hellman anahtar değişimi incelenmiştir.

3.2. Diffie-Hellman anahtar değişimi

Diffie-Hellman anahtar değişimi, simetrik şifrelemede oluşturulan anahtarın güvensiz ortamda güvenli bir şekilde paylaşımını sağlamıştır. Bu yöntem modüler üs alma teoremine dayanmaktadır (Yerlikaya vd., 2006).

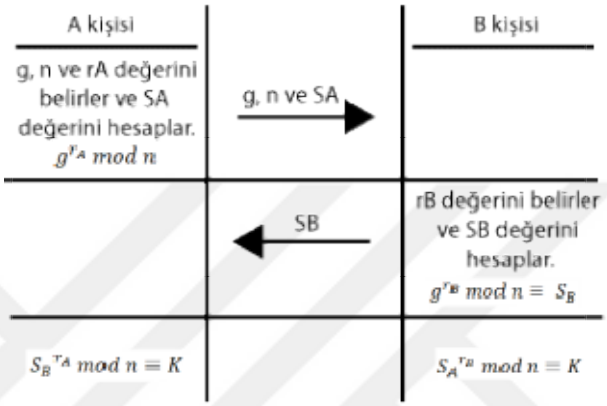
Anahtar oluşturacak kişiler A ve B olsun. A kişisi rastgele g ve n sayılarını belirler. g sayısı üssünü almak, n sayısı mod almak için kullanılacaktır. Yine aynı şekilde A kişisi rastgele r_A sayısı belirler. $g^{r_A} \, mod \, n \equiv \, S_A$ işlemini gerçekleştirerek S_A değerini hesaplar. A kişisi rastgele belirledi g, n sayılarını ve hesapladığı S_A sayısını B kişisine gönderir. B kişisi rastgele r_B sayısı belirler. $g^{r_B} \, mod \, n \equiv \, S_B$ işlemini gerçekleştirerek S_B değerini hesaplar. B kişisi hesapladığı S_B sayısını A kişisine gönderir. A kişisi ${S_B}^{r_A} \, mod \, n \equiv K$ işlemini, B kişisi ${S_A}^{r_B} \, mod \, n \equiv K$ işlemini gerçekleştirerek K değerlerini bulurlar. K değeri A ve B kişileri için oluşturdukları gizli simetrik anahtardır (Yerlikaya vd., 2006).

$${S_B}^{r_A} \, mod \, n \equiv {S_A}^{r_B} \, mod \, n \equiv K \tag{3.1}$$

(3.1)'de S_B yerine g^{r_B} ve S_A yerine g^{r_A} yazılabilir.

$$(g^{r_B})^{r_A} \, mod \, n \equiv (g^{r_A})^{r_B} \, mod \, n \equiv K \tag{3.2}$$

Bu yöntemin güvenilirliği açık olarak paylaşılan g, n, S_A ve S_B bilgilerinden gizli olan r_A ve r_B bilgilerine ulaşılmasının zor olmasından kaynaklanmaktadır. Bu fonksiyonlara tek yönlü fonksiyonlar veya trapdoor function (kapak fonksiyonu) denilmektedir. Yapılması kolay ama bazı bilgiler olmadan geri dönülmesi zordur. Kırılabilmesi için sadece deneme yanılma yöntemi kullanılabilir (Yerlikaya vd., 2006).

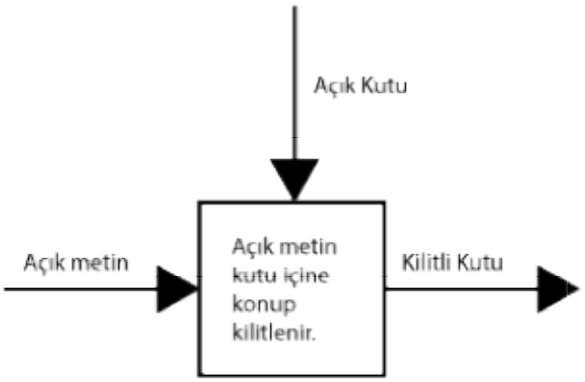


Şekil 3.1. Diffie-Hellman anahtar değişimi

Örnek 3.1: A kişisi $g = 10$, $n = 101$ ve $r_A = 5$ olarak seçsin. A kişisi $g^{r_A} \bmod n \equiv S_A = 10^5 \bmod 101 \equiv 1$, S_A değerini hesaplar ve $g, n \text{ ve } S_A$ değerlerini B kişisine gönderir. B kişisi $r_B = 7$ olarak rastgele bir sayı seçsin. B kişisi $g^{r_B} \bmod n \equiv S_B = 10^7 \bmod 101 \equiv 91$, S_B değerini hesaplar ve A S_B değerini A kişisine gönderir. İki taraf rastgele seçtikleri sayılar ile karşı taraftan gelen sayının üssünü alarak ortak anahtarlar değerlerini bulurlar; $S_B{}^{r_A} \bmod n \equiv S_A{}^{r_B} \bmod n \equiv K \equiv 91^5 \bmod 101 \equiv 10^7 \bmod 101 \equiv 91$.

3.3. RSA Algoritması

Simetrik anahtar şifrelemenin dezavantajlarından dolayı 1970'li yıllarda James H. Ellis simetrik şifrelemedeki gizli anahtar yerine açık anahtar ile şifreleme üzerine çalışmalar yapmıştır. Şekil 3.1 ve Şekil 3.2'de James H. Ellis'in ortaya attığı şifreleme ve şifre çözme fikirleri gösterilmiştir (Ural & Örenç, 2018, s. 59-61).



Şekil 3.2. James H. Ellis şifreleme fikri



Şekil 3.3. James H. Ellis şifre çözme fikri

James H. Ellis bu fikrini matematiksel olarak gerçekleştirememiş olmasına rağmen kriptografi dünyasına bir fikir tohumu ekmiştir. Bu fikir şifreleme ve şifre çözme işlemlerinde kullanılan anahtarların birbirinden farklı olması ama tersine çalışması yani asimetrik şifrelemedir. 1977 yılında Ron Rivest, Adi Shamir ve Leonard Adleman (Rivest vd., 1978) soyadlarının baş harfleri ile isimlendirdikleri RSA algoritmasını oluşturdular. RSA asimetrik şifreleme yapısına sahip bir kriptoloji algoritmasıdır ve günümüzde kullanılan asimetrik şifreleme algoritmaların geliştirilmesinde önemli rol oynamıştır (Forouzan, 2008).

RSA algoritması, tam sayıların çarpma işleminin kolay bir biçimde yapılmasına ve tam sayıların çarpanlarına ayrılmasının zorluğuna dayanmaktadır. Günümüzde bilgisayarların yardımı ile çarpma işlemlerini çok hızlı bir şekilde yapabiliyoruz. Fakat bilgisayarların yardımı ile bile çarpanlara ayırma işlemi kolay bir şekilde yapılamamaktadır ve çok büyük sayılarda bu işlem yıllar sürebilmektedir. R. Rivest, A.

Shamir ve L. Adleman çarpanlarına ayırma işleminin bu zorluğunu fark ederek çarpanlarına ayırma işlemini güvenlik olarak kullanarak RSA algoritmasını geliştirmişlerdir (Ural & Örenç, 2018, s. 59-61).

3.3.1. Algoritmanın Altyapısı

Açık metin m, şifreli metin c, d, e ve n değerleri eşitliği sağlayan değerler olsun.

$$m^e \ mod\ n = c,\ c^d \ mod\ n = m \tag{3.3}$$

(3.3)'deki eşitliği sağlayan d ve e değerlerinin üretiminin kolay olması için Euler phi fonksiyonundan yararlanılmıştır.

İki tane çok büyük asal sayı seçilir. Asal sayılar A_1 ve A_2, açık metin m ve şifreli metin c olsun.

$$A_1 \times A_2 = n \tag{3.4}$$

Euler phi fonksiyonuna göre;

$$\varphi(n) = (A_1 - 1) \times (A_2 - 1\ ) \tag{3.5}$$

Euler phi fonksiyonu ve modüler üs alma teoremi birleştirildiğinde;

$$z^{\varphi(n)} \equiv 1\ mod\ n \text{ (z denkliği sağlayan sayı)} \tag{3.6}$$

$$z^{k\times\varphi(n)+1} \equiv 1\ mod\ n \text{ (}k\text{herhangi bir sayı)} \tag{3.7}$$

$$d \times e = k \times \varphi(n) + 1 \tag{3.8}$$

$$d = \frac{k\times\varphi(n)+1}{e} \tag{3.9}$$

Bu eşitliğe göre (3.4)'e bakıldığında;

$$m^d \ mod\ n = c,\ c^e \ mod\ n = m \tag{3.10}$$

$$(m^d)^e \; mod \; n = m \tag{3.11}$$

$$m^{d \times e} \; mod \; n = m \tag{3.12}$$

Anahtar Üretimi: İki tane büyük asal sayı A_1, A_2olsun. $n = A_1 \times A_2$, $\varphi(n) = (A_1 - 1) \times (A_2 - 1)$'dir.

$\varphi(n)$ ile ortak bir çarpanı olmayan ve $1 < y < \varphi(n)$ koşulunu sağlayan rastgele bir y sayısı seçilir.

$d = \frac{k \times \varphi(n) + 1}{e}$ sonucu hesaplanarak d değeri bulunur.

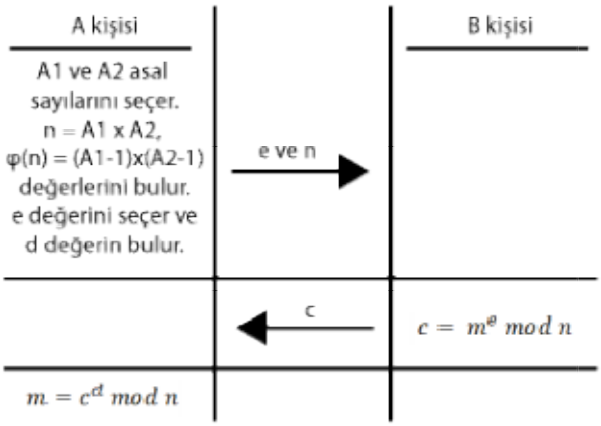
d gizli anahtar, e açık anahtardır. Şifreleme işlemi için e ve n değerleri paylaşılır, diğer değerler saklanır.

Şifreleme: m açık metin, c şifreli metin, e ve n değerleri açık anahtar değerleri olsun.

$$c = \; m^e \; mod \; n \tag{3.13}$$

Şifre Çözme: m açık metin, c şifreli metin, e ve n değerleri açık anahtar değerleri olsun.

$$m = c^d \; mod \; n \tag{3.14}$$



Şekil 3.4. RSA algoritması

Örnek 3.2: A_1 = 53 ve A_2 = 59 iki asal sayı seçilir.

$A_1 \times A_2 = n, 53 \times 59 = 3217, n = 3217$
$\varphi(n) = (A_1 - 1) \times (A_2 - 1), \varphi(3217) = 52 \times 58 = 3016$

$\varphi(3217)$ ile ortak bir çarpanı olmayan ve $1 < e < \varphi(n)$koşulunu sağlayan rastgele bir e sayısı seçilir, $e = 3$ olsun.

$d = \frac{k \times \varphi(n)+1}{e}$, $d = \frac{2 \times 3016+}{3} = 2011$ (k değeri d değerini tamsayı olmasını sağlayacak katsayıdır)

A_1, A_2, x ve $\varphi(n)$ değerleri gizli tutulacak, n ve e değerleri açık bir şekilde paylaşılacaktır. Açık metin m ve şifreli metin c olsun.

Şifreleme:

$m = 89$ olsun.
$m^e \ mod \ n = c$
$89^3 \ mod \ 3217 = 1394$

Şifre çözme:

$c^d \ mod \ n = m$
$1394^{2011} \ mod \ 3217 = 89$'dur.

3.3.2. RSA Algoritmasına Karşı Saldırılar

Şimdiye kadar RSA algoritmasına karşı güçlü bir saldırı bulunamadı. Açık metin zayıflığı, parametrelerin seçimindeki zayıflık öngörülen saldırılardır (Forouzan, 2008).

Çarpanlara Saldırı (Factorization Attack): RSA algoritmasının güvenliği, modülün çarpanlara ayrılmasının uzun sürmesine dayanmaktadır. Fakat saldırgan bir şekilde modül değerini çarpanlarına ayırıp gizli olan asal sayılara ulaşabilirse, her şifrelenmiş metini çözebilir. Bu saldırı RSA algoritmasının çarpma özelliğine

dayanmaktadır. B kişisi A kişisine şifrelenmiş metin (c) değerini gönderirken Z kişisi tarafından bu bilgi ele geçirilebilir. Z kişisi sırasıyla alt taraftaki işlemleri gerçekleştirerek açık metini elde eder;

- Z kişisi rastgele bir sayı seçer, o.
- $v = c \times o^e \; mod \; n$
- v değerini A kişisine göndererek şifre çözme işleminin sonucu elde eder, s.
- $s = v^d \; mod \; n = \; (c \times o^e)^d \; mod \; n = (c^d \times o^{ed}) \; mod \; n = (c^d \times o) \; mod \; n = (m \times o) \; mod \; n$
- $m = s \times o^{-1} \; mod \; n$

Bu saldırı bölüm 3.4'te anlatılan ElGamal algoritmasına benzer olarak uzatılmış Öklid algoritmasını kullanmaktadır (Forouzan, 2008).

Şifreleme Üs Değerine Saldırı (Attacks on The Encryption Exponent): RSA algoritmasında kullanılan açık anahtarın işlem zamanını azaltmak için küçük seçilmesinden kaynaklanır. Bu saldırı genellikle sistemin bozulmasına neden olmaz, ancak yine de önlenmesi gerekir. Bu saldırıları önlemek için çok küçük açık anahtar değerleri seçilmemesi gerekmektedir. Bu saldırı Franklin Reiter tarafından keşfedilmiştir. A kişisi m_1 ve m_2 diye 2 tane açık metini y = 3 açık anahtarı ile üssünü alarak şifreler vec_1 ve c_2 şifreli metinleri gönderir. Eğer m_1 ve m_2 açık metinleri birbiri ile ilişkili ise, belirli bir süre ile saldırgan açık metinleri elde edebilir (Forouzan, 2008).

Şifre Çözme Üs Değerine Saldırı (Attacks on the Decryption Exponent): Bu saldırı gizli olan şifre çözme üs değerini saldırgan tarafından ele geçirilmesi ile gerçekleştirilir (Forouzan, 2008).

Açık Metin Saldırısı (Plaintext Attacks): RSA algoritmasındaki açık metin ve şifreli metin, aynı aralıktaki tam sayılar oldukları için birbirlerinin permütasyonlarıdır. Yani, saldırgan açık metin hakkında bazı bilgilere sahipse, açık metin üzerinde bazı saldırılara izin verebilir. Eğer saldırgan olası açık metinler kümesini bilirse, sonuç, ele geçirilen şifreli metinle aynı olana kadar tüm olası mesajları şifreleyebilir. Örneğin, A kişisinin B kişisine dört basamaklı bir sayı gönderdiği biliniyorsa, saldırgan, açık metni bulmak için kolayca 0000 ile 9999 arasındaki numaraları deneyebilir. Bu nedenle, kısa mesajların önündeki rastgele bitlerle doldurulmalıdır (Forouzan, 2008).

Modül Değerine Saldırı: RSA algoritmasına gerçekleştirebilecek ana saldırı olan çarpanlara saldırının farklı bir çeşididir. Küçük değerdeki modül değerine yapılan bir saldırıdır. Buna ek olarak yaygın olarak kullanılan modüllerin seçilmesinden kaynaklı saldırılarda gerçekleştirilebilir (Forouzan, 2008).

3.4. ElGamal Algoritması

Anahtar değişim problemi ana şifreleme zorluklarından biridir. Taher Elgamal 1984 (ElGamal, 1984) yılında asimetrik bir şifreleme algoritması önerdi. El-gamal algoritması, ele geçirilmiş bir anahtarın olma olasılığını atlamak için kullanılan ortak bir anahtar şifreleme sistemidir. Diffie-Hellman anahtar değişimi ve uzatılmış öklid algoritması yöntemine dayanmaktadır ve anahtar her seferinde değişmektedir (Ural & Örenç, 2018, s. 72-73).

3.4.1. Algoritmanın Altyapısı

Bilgi paylaşacak kişiler A ve B olsun. A kişisi rastgele x ve y sayılarını belirler. x sayısı üssünü almak, y sayısı mod almak için kullanılacaktır. Yine aynı şekilde A kişisi rastgele r_A sayısı belirler. $x^{r_A} \, mod \, y \equiv \, S_A$ işlemini gerçekleştirerek S_A değerini hesaplar. A kişisi rastgele belirlediği x, y sayılarını ve hesapladığı S_A sayısını B kişisine gönderir, r_A gizli anahtardır ve saklanır. B kişisi rastgele r_B sayısı belirler. $x^{r_B} \, mod \, y \equiv \, S_B$ işlemini gerçekleştirerek S_B değerini hesaplar. B kişisi ${S_A}^{r_B} \, mod \, y \equiv K$ işlemini gerçekleştirerek K değerlerini bulurlar. B kişisinin göndermek istediği mesaj m ve şifrelenmiş metin c olsun. B kişisi $K \times m \, \equiv c \, mod \, N$ işlemini gerçekleştirerek şifreleme işlemini gerçekleştirir. B kişisi hesapladığı c ve S_B değerlerini A kişisine gönderir. A kişisi ${S_B}^{r_A} \, mod \, y \equiv K, \;\; K^{-1} \, mod \, y = L$ işlemlerini gerçekleştirerek L değerini hesaplar. A kişisi $L \times c \, mod \, y \, \equiv m$ işlemini gerçekleştirerek mesaja ulaşmış olur (Ural & Örenç, 2018, s. 72-73).

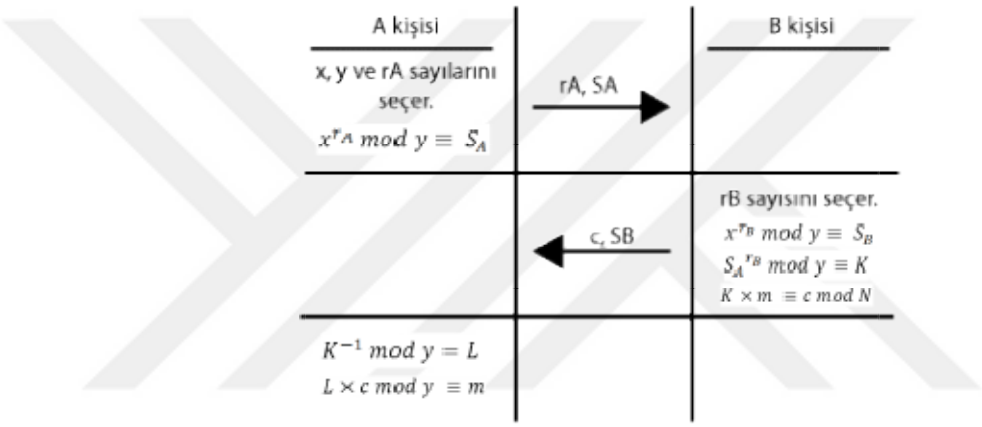
Anahtar Üretimi: Rastgele x, y ve r_1 sayı bilgi alacak taraf tarafından belirlenir. $S_1 \equiv \, x^{r_A} \, mod \, y$ değeri hesaplanır ve karşı tarafaS_1, x ve y değerleri gönderilir. Karşı taraf rastgele r_2 sayısını belirler. ${S_1}^{r_2} \, mod \, y \equiv K$ değerini hesaplar. K değeri ortak gizli anahtar ve$K^{-1} \, mod \, y \equiv L$ değeri K anahtarının asimetrik anahtar karşılığıdır.

Şifreleme:

$$K \times m \equiv c \; mod \; N \tag{3.15}$$

Şifre Çözme:

$$L \times c \; mod \; y \equiv m \tag{3.16}$$



Şekil 3.5. ElGamal algoritması

Örnek 3.3:

Bilgi paylaşacak kişiler A ve B, $x = 5$, $y = 89$, A kişisinin rastgele seçtiği sayı $r_A = 13$ ve B kişisinin seçtiği rastgele sayı $r_B = 8$olsun.

A kişisi S_A değerini hesaplayarak, x,y ve S_A değerlerini B kişisin gönderir.

$x^{r_A} \; mod \; y \equiv \; S_A$, $5^{13} \; mod \; 89 \equiv \; 40$

B kişisinin mesajı $m \; = \; 35$ olsun. B kişisi şifreleme işlemlerini yaparak c ve S_B değerlerini A kişisine gönderir.

$x^{r_B} \; mod \; y \equiv \; S_B$, $5^8 \; mod \; 89 \equiv \; 4$

$S_A{}^{r_B} \; mod \; y \equiv K$, $40^8 \; mod \; 89 \equiv 16$

$K \times m \; \equiv c \; mod \; N$, $16 \times 35 \; \equiv 26 \; mod \; 89$

A kişisi şifre çözme için;

$S_B{}^{r_A}\ mod\ y \equiv K$, $4^{13}\ mod\ 89 \equiv 16$

$K^{-1}\ mod\ y = L$, $16^{-1}\ mod\ 89 = 39$

$L \times c\ mod\ y\ \equiv m$, $39 \times 26\ mod\ 79\ \equiv 35$ olarak mesaja ulaşır.

3.4.2. ElGamal Algoritmasına Karşı Saldırılar

ElGamal algoritmasına literatürde geçen iki saldırı çeşidi vardır. Küçük modül saldırısı (low-modulus attacks) ve bilinen açık metin saldırısı (known-plaintext attack)'dır (Forouzan, 2008).

3.5. Paillier Algoritması

Pailler algoritması 1999 yılında Pascal Pailler (Pailler, 1999) tarafından asimetrik şifreleme algoritması olarak sunuldu. n dereceli kök sınıflarınını hesaplamasının zorluğunu dayanmaktadır (Pettersen, 2016, s. 42-67).

3.5.1. Algoritmanın Altyapısı

Bilgi paylaşacak kişiler A ve B olsun. A kişisi$x$ ve y olarak eşit uzunlukta iki tane büyük asal sayı seçer. $n = x \times y$,$u = EBOB\big((x-1),(y-1)\big)$,$g = n + 1$ olarak hesaplanır.A kişisi n ve g değerlerini açık olarak paylaşır ve u ve a değerlerini gizli anahtar olarak saklar. Şifrelenmek istenilen mesaj m olsun,$m \in Z_n$ olmalıdır. B kişisi şifrelemek istediği mesajı şifrelemek için $r \in Z_n$ koşulunu sağlayacak rastgele bir r sayısı seçer. $(g^m \times r^n)\ mod\ n^2 = c$ şeklinde mesajını şifreler ve A kişisine gönderir. A kişisi mesaja ulaşmak için $\frac{L(c^u\ mod\ n^2)}{L(g^u\ mod\ n^2)}\ mod\ n$ işlemini gerçekleştirerek mesaja ulaşır(L fonksiyonu $L(a) = \frac{a-1}{n}$ şeklindedir) (Pettersen, 2016, s. 42-67).

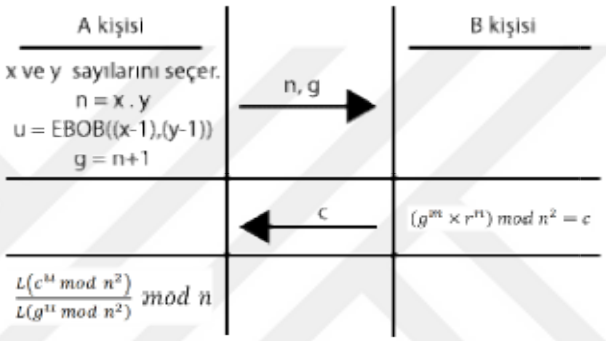
Anahtar Üretimi: Rastgele x ve$y$eşit uzunlukta iki büyük asal sayı bilgi alacak taraf tarafından belirlenir. $n = x \times y$, $u = EBOB\big((x-1),(y-1)\big)$ ve $g = n + 1$ değerleri hesaplanır. n ve g değerlerini açık anahtar, u ve a değerlerini gizli anahtardır.

Şifreleme:

$$c = (g^m \times r^n)\ mod\ n^2 \tag{3.17}$$

Şifre Çözme:

$$m = \frac{L(c^u\ mod\ n^2)}{L(g^u\ mod\ n^2)}\ mod\ n \tag{3.18}$$



Şekil 3.6. Paillier algoritması

Örnek 3.3:

Bilgi paylaşacak kişiler A ve B olsun. A kişisi $x = 5$ ve $y = 7$ olarak iki tane eşit uzunlukta asal sayı seçer. A kişisi

$n = 5 \times 7 = 35$,

$u = EBOB((5-1) \times (7-1)) = 12$,

$g = 35 + 1 = 36$ ve

$a = 24^{-1}\ mod\ 35 = 19$olarak hesaplar.

(n, g) değerlerini açık olarak paylaşır ve (u, a) değerlerini gizli anahtar olarak saklar.

B kişisinin şifrelemek istediği mesaj $m = 1$, rastgele seçtiği sayı $r = 4$ve şifreli metin $c$olsun. B kişisi aşağıdaki işlemi gerçekleştirerek,

$(g^m \times r^n)\ mod\ n^2 = c$, $(36^1 \times 4^{35})\ mod\ 35^2 = c$,$c = 639$

şifrelediği değeri A kişisine gönderir.

A kişisi şifre çözme işlemini gerçekleştirerek mesaja ulaşır.

$\frac{L(c^u \bmod n^2)}{L(g^u \bmod n^2)} \bmod n$, $\frac{L(639^{12} \bmod 1225)}{L(36^{12} \bmod 1225)} \bmod 35 = 1$

BÖLÜM 4

HOMOMORFİK ŞİFRELEME

4.1. Giriş

Şifreleme işlemi bir bilgiyi bir yerden bir yere güvenli bir biçimde aktaralılmasını sağlamanın yanında bilginin güvenli bir şekilde saklanmasınıda sağlamaktadır. Bilgi saklama, bilgiyi göndermede olduğu gibi bilginin bir anahtar ile şifrelenmesi ve bir alanda saklanması şeklinde gerçekleşir. Anahtara sahip kullanıcılar veya günümüzde çok fazla kullanımı olan bulut hizmetlerinide kapsayan servis sağlayıcıları bilgiler üzerinde özel haklara sahiptir. Anahtarın doğrudan bir paylaşımı olmasa bile, bilgi, üçüncü bir taraf ile yapılması gereken bir işlem için paylaşılması gerekebilir. Tam bu noktada bilginin güvenliğine karşı bir zafiyet oluşmaktadır. Homomorfik Şifreleme, bilgi üzerinde yapılmak istenilen matematiksel işlemi şifrelenmiş metin üzerinde yapılmasını olanak sağlayan bir kriptografik yöntemdir. Bu sayede, bilgiler üzerinde işlemler güvenli bir şekilde yapılabilir (Acar vd., 2018, s. 4-5).

Bilgiyi bozmadan, şifrelenmiş veriler üzerinde basit işlemlerin yapılabilmesi düşüncesi ilk olarak 1978 yılında Rivest, Adleman ve Derouzous tarafından ortaya atılmıştır. 1978'den bu yana bu konu önem kazanmış ve bu konuda sayısız çalışma yapılmıştır (Sen, 2013, s. 1).

2009 yılına kadar RSA, ElGamal ve Pailler algoritmaları gibi birçok algoritmada uygulanmış, fakat sadece tek bir işlem tipi yapılabilmesine olanak sağlamıştır. 2009 yılında Craig Gentry şifreli veriler üzerinde birden fazla işlem yapabilme yani Tam-Homomorfik Şifreleme fikrini yayınlaması ile bu konu tekrardan önem kazanmıştır. Bunun büyük bir başarı olmasına karşın,bu konunun iyileştirilmesi, uygulanması ve var olan uygulamalar ile uyumlu hale getirilmesi gerektiği gösterilmiştir (Acar vd., 2018).

Tezin bu bölümünde Homomorfik Şifreleme ve çeşitleri olan Kısmi-HE, Somewhat HE (SWHE) ve Tam-HE yöntemleri incelenmiştir.

4.2. Homomorfik Şifreleme

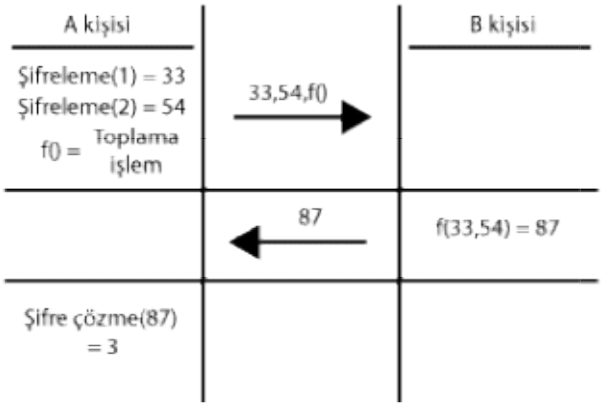
Şifreleme bilgilerin güvenliğini sağlamak için çok önemlidir. Fakat geleneksel şifreleme yöntemleri bilgiler üzerinde yapılacak işlemleri, şifre çözmeden gerçekleştiremez. Bunun yanı sıra, günümüzde yaygın olarak kullanılan bulut sistemleri ile bilgilerin depolanması ve paylaşılması gizliliğimizden ödün vermemizi gerektirmektedir. Bilgilerin hem rahatça paylaşılması hem de güvenliğinin sağlanması için şifrelenmiş bilgiler üzerinde işlemlerin yapılabilmesine olanak sağlayacak bir yapının olması gerekmektedir (Sen, 2013).

Özet olarak bu yapının gereksinimlerini şöyle sıralayabiliriz;

- Açık metin üzerinde değil şifrelenmiş metin üzerinde matematiksel işlemler yapılmalıdır.
- Şifreli metnin şifresi çözüldüğünde elde edilen sonuç, aynı işlemlerin açık metin üzerinde yapılması ile elde edilecek sonuç ile aynı olmalıdır.

Bu gereksinimleri sağlayan yapılara Homomorfik Şifreleme yapıları denilmektedir.

Örnek 4.1: A kişisi 1'e 2'yi eklemek istiyor fakat sayıların nasıl ekleneceğini bilmiyor. A bu problemi çözmek için başka birinden yani B kişisinden yardım almak istiyor. Fakat A kişisi elinde bulunan 1 ve 2'yi güvenlik nedeni ile herhangi biri ile paylaşmakta istemiyor. Bundan dolayı A kişisi yardım almak istediği B kişisine aynı zaman güvenmemektedir. A kişisi hem 1'e 2'yi eklemek için B kişisinden yardım alıp hem de B kişisine hangi sayıları eklemek istedi bilgisini vermeden bu işlemi gerçekleştirmek için önce elindeki sayıları (1,2) şifreleme işlemi gerçekleştirerek yeni sayılar elde eder. Şifreleme(1) = 33, Şifreleme(2) = 54. A kişisi elde ettiği bu sayıları ve yapmak istediği toplama işlemini (f) B kişisine gönderir. B kişisi sadece şifrelenmiş olan 33 ve 54 sayılarını bilmektedir. B kişisi bu sayıları toplar 33 + 54 = 87 ve sonucu A kişisine geri gönderir. A kişisi gelen cevabı şifre çözme işlemi gerçekleştirerek Şifre çözme (87) = 3 cevabına ulaşmış olur.



Şekil 4.1. Örnek gösterimi

Yani HE, şifreli metin üzerinde işlem gerçekleştirebileceğiniz ve elde edeceğiniz bu sonucun şifresi çözüldüğü zaman, açık metin ile bu işlemi gerçekleştirdiğiniz sonuç ile aynı olacaktır.

HE formül ile ifade etmek gerekirse; $Şif_k()\ k$ anahtarı ile şifreleme işlemini, $ŞifÇöz_k()k$ anahtarı ile şifre çözme işlemini, k anahtarı, **(+)** ve **(*)** işaretleri Z kümesi üzerindeki toplama ve çarpma işlemlerini temsil etsin ve $a, b \in Z$ değerleri olsun. Bu durumda (Çalık, Erdem & Aydın, 2015);

HE toplama özelliğini taşıması için;

$$a + b = ŞifÇöz_k(Şif_k(a) + Şif_k(b)) \quad (4.1)$$

HE çarpma özelliğini taşıması için;

$$a * b = ŞifÇöz_k(Şif_k(a) * Şif_k(b)) \quad (4.2)$$

eşitliğini sağlaması gerekmektedir.

Bu sayede bilgilerin gizliliği sürdürülerek güvenli bir şekilde bilgiler üzerinde matematiksel işlemler yapılabilmektedir.

Homomorfik Şifreleme, yapılacak işlem sayısına göre Kısmi-Homomorfik Şifreleme ve Tam-Homomorfik şifreleme olarak iki kısma ayrılmaktadır (Çalık vd., 2015).

4.3. Kısmi-Homomorfik Şifreleme

Kısmi-Homomorfik Şifreleme yönteminde ya çarpma işlemi ya da toplama işlemi olmak üzere sadece bir tek işlem gerçekleştirebilir. RSA, ElGamal ve Paillier Kısmi-Homomorfik Şifreleme özelliği göstermektedir.

4.3.1. RSA Algoritması ve HE

RSA algoritması bilinen ilk Homomorfik Şifreleme yöntemlerinden biridir. Çarpmaya göre Kısmi-Homomorfik Şifreleme özelliği gösterir (Parmar vd, 2014). m_1 1. açık metin, m_2 2. açık metin, n mod değeri, y açık anahtar olsun. RSA algoritmasının çarpmaya göre Kısmi-Homomorfik Şifreleme özelliğini aşağıdaki gibi gösterilebilir;

$$m_1^y \; mod\; n \;\times\; m_2^y \; mod\; n = \; m_1^y \times m_2^y \; mod\; n = (m_1 \times m_2)^y \; mod\; n \qquad (4.3)$$

(4.3)'de görüldüğü gibi iki açık metinin şifrelenmiş olarak çarpımı açık metin olarak çarpımının şifrelenmiş haline eşittir.

Örnek 4.2: $m_1 = 4$ ve$m_2 = 3$ açık metin değerleri, $A_1 = 3$ ve $A_2 = 11$ asal değerler, $n = \; A_1 \times A_2 = 33$ mod değeri, $\varphi(n) = (A_1 - 1) \times (A_2) = 20, y = 7$ açık anahtar, $x = \frac{k \times \varphi(n)+1}{y} = \frac{1 \times 20+1}{7} = 3$ gizli anahtar, c_1 ve c_2 şifreli metin değerleri olsun.

$c_1 = m_1^y (mod\; n) = \; 4^7 (mod\; 33) = 16$
$c_2 = m_2^y (mod\; n) = 3^7 (mod\; 33) = 9$
$c_1 \times c_2 = 16 \times 9 = 144$
$(c_1 \times c_2)^x (mod\; n) = \; (144)^x (mod\; 33) = 12$
$m_1 \times m_2 = 3 \times 4 = 12$
$(c_1 \times c_2)^x (mod\; n) = \; m_1 \times m_2 = 12$

4.3.2. ElGamal Algoritması ve HE

ElGamal algoritması toplamaya göre Homomorfik Şifreleme özelliği göstermektedir (Parmar vd, 2014). m_1 1. açık metin, m_2 2. açık metin, y mod değeri,

Kortak anahtar olsun. ElGamal algoritmasının toplamaya göre Kısmi-Homomorfik Şifreleme özelliğini aşağıdaki gibi gösterilebilir;

$$(K \times m_1)\ mod\ y + (K \times m_2)\ mod\ y = (K \times m_1) + (K \times m_2)\ mod\ y = K(m_1 + m_2)\ mod\ y \quad (4.4)$$

(4.4)'de görüldüğü gibi iki açık metinin şifrelenmiş olarak toplamı açık metin olarak toplamının şifrelenmiş haline eşittir.

Örnek 4.3: $m_1 = 4$ ve $m_2 = 3$ açık metin değerleri, $x = 5$ sayısı üssünüalma değeri, $y = 89$ sayısı mod değeri, $r_A = 13$ rastgele sayı, $r_B = 8$ rastgele sayı, $S_A = x^{r_A}\ mod\ y = 40$, $S_B = x^{r_B}\ mod\ y = 4$, $K = {S_A}^{r_B}\ mod\ y = {S_B}^{r_A}\ mod\ y = 16$ ortak anahtar, $L = K^{-1}\ mod\ y = 39$, c_1 ve c_2 şifreli metin değerleri olsun.

$c_1 = K \times m_1\ mod\ y = 16 \times 4\ mod\ 89 = 64$
$c_2 = K \times m_2\ mod\ y = 16 \times 3\ mod\ 89 = 48$
$c_1 + c_2 = 64 + 48 = 112$
$(c_1 + c_2) \times L\ mod\ y = (112) \times 39\ mod\ 89 = 7$
$m_1 + m_2 = 3 + 4 = 7$
$(c_1 + c_2) \times L\ mod\ y = m_1 + m_2 = 7$

4.3.3. Paillier Algoritması ve HE

Paillier algoritması toplamaya göre Homomorfik Şifreleme özelliği göstermektedir (Parmar vd, 2014).Şifrelenmiş metinlerin toplamı, açık metinlerin toplamına eşittir. m_1 1. açık metin, m_2 2. açık metin, n mod değeri, g açık anahtar, r seçilen rastgele değer olsun. Paillier algoritmasının toplamaya göre Kısmi-Homomorfik Şifreleme özelliğini aşağıdaki gibi gösterilebilir;

$$(g^{m_1} \times r^n)\ mod\ n^2 \times (g^{m_2} \times r^n)\ mod\ n^2 = (g^{m_1} \times r^n) \times (g^{m_2} \times r^n)\ mod\ n^2 = r^{2n} g^{m_1+m_2}\ mod\ n^2 \quad (4.5)$$

(4.5)'de görüldüğü gibi iki açık metinin şifrelenmiş olarak toplamı açık metin olarak toplamının şifrelenmiş haline eşittir.

Örnek 4.4: $m_1 = 4$ ve $m_2 = 3$ açık metin değerleri, $x = 5$ ve $y = 7$ asal değerler, $n = x \times y = 35$ sayısı mod değeri, $u = EBOB((x-1) \times (y-1)) = 12$,$S_A = x^{r_A} \, mod \, y = 40$,$g = n + 1 = 36$,$r = 4$ rastgele sayı, c_1 ve c_2 şifreli metin değerleri olsun.

$$c_1 = (g^{m_1} \times r^n) \, mod \, n^2 = (36^4 \times 4^{35}) \, mod \, 1225 = 359$$

$$c_2 = (g^{m_2} \times r^n) \, mod \, n^2 = (36^3 \times 4^{35}) \, mod \, 1225 = 44$$

$$c_1 \times c_2 = 359 + 44 = 15796$$

$$\frac{L((c_1 \times c_2)^u \, mod \, n^2)}{L(g^u \, mod \, n^2)} \, mod \, n = \frac{L(15796^{12} \, mod \, 1225)}{L(36^{12} \, mod \, 1225)} \, mod \, 35 = \frac{L(490)}{L(421)} \, mod \, 35$$

$$= \frac{14}{12} \, mod \, 35 = 7$$

$$m_1 + m_2 = 3 + 4 = 7$$

$$\frac{L((c_1 \times c_2)^u \, mod \, n^2)}{L(g^u \, mod \, n^2)} \, mod \, n = m_1 + m_2 = 7$$

4.3. Homomorfik Benzeri Şifreleme Sistemleri

SWHE yapıları Kısmi-HE yapılarından farklı olarak hem toplama hem de çarpma işlemlerini bir arada gerçekleştirebilir. Fakat bu işlemleri sınırsız sayıda gerçekleştiremez. SWHE yapıları FHE yapılarının gelişiminde önemli bir rol oynamıştır. FHE yapıları için bir yapı taşı olarak düşünülebilir (Ogburn, Turner & Dahal, 2013).

SWHE yapısı 1982 yılında A.C. Yao (Yao, 1982), 1999 yılında T. Sander, A. Young ve M. Yung (Sander, Young, Yung, 1999), 2005 yılında D. Boneh, E. J. Goh ve K. Nissim (Boneh, Goh, Nissim 2005), 2007 yılında Y. Ishai ve A. Paskin (Ishai & Paskin, 2007) tarafından FHE yapısından önce literatürde yerini almıştır (Acar vd., 2018, s. 9).

4.3.1. Boneh-Goh-Nissim (BGN) Şifreleme

2005 yılında Boneh, Goh ve Nissim tarafından SWHE yapısı önerilmiştir. BGN yapısı açık anahtar yapılıdır. BGN şifreleme yöntemi sınırsız sayıda toplama işlemine ve bir tane çarpma işlemine izin vermektedir. Pailier algoritması ile benzerlik göstermektedir (Lai, Zhou & Li, 2011).

Anahtar Üretimi: $s \in Z^+$, s güvenlik parametresi olarak belirlenir. s-bit uzunluğuna sahip iki tane A_1 ve A_2 asal sayıları seçilir. n değeri $n = A_1 \times A_2$ şeklinde hesaplanır. n mertebesinden G_1 ve G_2 çarpım işlemine göre iki grup üretilir. Bilinear haritası $e : G_1 \times G_1 \rightarrow G_2$ şeklindedir (Tüm x, $y \in G_1$ ve a, $b \in Z$, $e(x^a, y^b) = e(x, y)^{ab}$)) ($g$'nin G_1 grubunun üreteci olması durumunda, $e(g, g)$ G_2 grubunun üreteci olur). g, $u \in_R G_1$ iki tane rastgele üreteç belirlenir. $h = u^{A_2}$ şeklinde A_1 mertebesinden G_1 grubunun alt grubundan bir üreteç hesaplanır. A_1 gizli anahtar ve (n, G_1, G_2, e, g, h) açık anahtardır (Lai vd., 2011).

Şifreleme: m açık metin, c şifreli metin olsun. $r \in_R \{0, \dots, n-1\}$ rastgele bir sayı seçilir (Lai vd., 2011).

$$c = g^m h^r \; mod \; n \tag{4.6}$$

Şifre Çözme:

$c' = c^{A_1} = (g^m h^r)^{A_1} = (g^{A_1})^m$ ve $g' = g^{A_1}$ ayrık logaritma ile mesaja ulaşılır (Lai vd., 2011);

$$m = log_{g'} c' \tag{4.7}$$

Ayrık logaritmanın hesaplanmasının hızlı olabilmesi için mesaj metin uzayının küçük tutulması gerekmetedir.

Homomorfik Toplam: m_1 ve m_2 açık metinler, c_1 ve c_2 şifreli metinler olsun (Acar vd., 2018, s. 9).

$$c_1 c_2 h^r = (g^{m_1} h^{r_1})(g^{m_2} h^{r_2}) h^r = g^{m_1 + m_2} h^{r'} \tag{4.8}$$

Homomorfik Çarpım: g_1, n mertebesinden ve h_1, g_1 mertebesindendir. $g_1 = e(g, g)$, $h_1 = g^{\alpha A_2}$. m_1 ve m_2 açık metinler, c_1 ve c_2 şifreli metinler olsun (Acar vd., 2018, s. 9).

$$e(c_1, c_2) h_1^r = e(g^{m_1} h^{r_1}, g^{m_2} h^{r_2}) h_1^r = g_1^{m_1 m_2} h_1^{m_1 r_2 + r_2 m_1 + \alpha q_2 r_1 r_2 r} = g_1^{m_1 m_2} h_1^{r'} \tag{4.9}$$

Şifreli metin çarpma işleminde G_1 grubundan G_2 grubuna geçmiştir. Bundan dolayı, G_1 grubundan başka bir çarpma işlemi yapılamaz. Fakat toplama işleminde

böyle bir durum söz konusu olmadığı için sınırsız sayıda toplama işlemi gerçekleştirilebilir (Acar vd., 2018, s. 9).

4.4. Tam-Homomorfik Şifreleme

Tam-Homomorfik Şifreleme, şifreli veriler üzerinde yapılan işlemler tek bir işlem tipiyle sınırlı kalmadan, farklı işlem tiplerini sayısız sayıda yapmaya izin veren yapılardır (Acar vd., 2018). Tam-Homomorfik Şifreleme yapısı temelde halka homomorfizmi olarak düşünülebilir (Yi, Paulet & Bertino, 2014).

A açık metin uzayı, B şifreli metin uzayı, $Şif()$şifreleme algoritması, ŞifÇöz() şifre çözme algoritması olsun. Açık metinler uzayı A'nın bir halka olduğunu (A, $\oplus_A$, $\otimes_A$) ve şifreli metinler uzayı B'nın bir halka olduğunu (B, $\oplus_B$, $\otimes_B$)varsayalım. Buna göre şifreleme işlemini A halkasından B halkasına tanımlı fonksiyon olarak düşünebiliriz (Yi vd., 2014).

$$Şif() : A \rightarrow C$$

$m_1, m_2 \in A$ olsun;

$$Şif(m_1) \oplus_B Şif(m_2) = Şif(m_1 \oplus_A m_2) = ŞifÇöz\big(Şif(m_1 \oplus_A m_2)\big) = m_1 + m_2 \quad (4.6)$$

$$Şif(m_1) \otimes_B Şif(m_2) = Şif(m_1 \otimes_A m_2) = ŞifÇöz\big(Şif(m_1 \otimes_A m_2)\big) = m_1 \times m_2 \quad (4.7)$$

şeklinde Tam-Homomorfik Şifreleme yapısı gösterilebilir (Yi vd., 2014).

2009 yılında, Gentry ilk Tam-Homomorfik Şifreleme yapısını açıkladı. Bu yapı Kısmi-Homomorfik ve Homomorfik Benzeri Şifrelemelerinin aksine hem toplama hem çarpma işlemlerini bir arada ve sınırsız sayıda yapmayı destekliyordu. Gentry'nin sunduğu yapı sadece bir Tam-Homomorfik Şifreleme yapısı olmasından bir yana, Tam-Homomorfik Şifreleme yapısı oluşturabilmek için nasıl bir yol izlenmesi gerektiğini anlatıyordu. Gentry'nin sunduğu yapıdan sonra birçok kişi yeni bir Tam-Homomorfik yapı tasarlamaya çalıştı (Acar vd., 2018).

Gentry'nin önerdiği yapı ideal kafes (ideal lattice) tabanlı bir yapıydı. Gentry'nin sunduğu Tam-Homomorfik Şifreleme yapısından öncede kafes tabanlı kriptoloji üzerinde çalışmalar devam etmekteydi. Bu yapı teorik olarak çok önemli bir yeri olsa da, uygulanabilirliği konusunda hesaplama maliyeti gibi sebeplerden dolayı

çok verimli değildi. Bu yapının daha verimli bir hale getirilmesi için birçok çalışma yapılmıştır.Yapılan yeni çalışmaların çoğu kafes problemlerindeki zorluğu güvenlik olarak kullanmasına dayanmaktadır (Acar vd., 2018).

Gentry, ideal kafeslere dayanan SWHE bir yapı kurarak başlamıştır. Fakat belirli bir işlem sayısından sonra elde edilen sonuçlar başarısız olmuştur. Bunun sebebi şifreli metindeki gürültü miktarının fazla olmasından kaynaklıdır (Acar vd., 2018). Gentry, blue print method denilen, Önyükleme (boostrapping) ve şifre çözme devresini önyükleme yapılabilir hale getirmek için bir teknik (squashing) işlemlerini önerdi. Boostrapping ve squashing işlemleri tekrar edebilir özelliktedir. Bundan dolayı oluşturulan yapı bu işlemlere sokularak sınırsız sayıda toplama ve çarpma işlemi gerçekleştirebilecektir (Gentry, 2009).

Squashing: Toplamı gizli anahtarın çarpımının tersine eşit olan bir vektör kümesi seçilir. Eğer şifreli metin bu kümenin elemanları ile çarpılırsa devrenin polinom derecesi şemanın kaldırabileceği seviyeye indirgenir. Şifreli metin artık boostrapping yapılabilir bir durumdadır (Acar vd., 2018).

Boostrapping: Şifrelenmiş metin içerisindeki gürültüyü ortadan kaldırmak için şifreli metini yeniden şifreleme işlemidir. Squashing işlemi gerçekleştirilerek boostrapping yapılabilir bir duruma gelen şifreli metin boostrapping işlemi gerçekleştirilerek yeni şifreli metin elde edilir (Gentry, 2009).

İlk önce iki farklı gizli ve açık anahtar çifti (aa1, ga1) ve (aa2,ga2) üretilir, açık anahtar ve gizli anahtar şifrelenir $a1 = Şif_{aa}\ (ga1)$ karşı taraf ile paylaşılır ve gizli anahtar saklanır. Karşı taraf mesajı, m, şifreler $c1 = Şif_{aa}\ (m)$. Gürültülü olan şifreli metin c1, şifrelenmiş olarak gelen anahtarı kullanarak $a1$ şifre çözme işlemini gerçekleştirir. Sonuç açık olarak paylaşılan diğer anahtar ile şifrelenir $Şif_{aa2}\big(ŞifÇöz_{a1}(c1)\big) = Şif_{aa}\ (m) = c2$ ve geri gönderilir. Gizli tutulan ga2 anahtarı ile şifre çözme işlemi gerçekleştirilerek sonuca ulaşılır $ŞifÇöz_{ga2}(c2) = m$ (Acar vd., 2018).

Yani, gürültü içeren şifreli metni şifre çözme işlemi uygulayarak gürültüyü kaldırır ve sonradan tekrar şifreleme işlemi gerçekleştiğinde küçük gürültülü bir hale getirir. Bu işlem gürültünün eşik noktasına ulaşana kadar tekrarlanabilir. Gentry'nin sunduğu bu yapıda maliyetin çok fazla olduğu görülmektedir. Bu şekilde bir SWHE

yapısı oluşturup, squashing ve boostrapping işlemleri gerçekleştirerek bir FHE yapısı oluşturulabilir (Acar vd., 2018).

4.4.1. İdeal Kafes Tabanlı FHE

Gentry'nin 2009 yılında açıkladığı yapı ideal kafes tabanlı bir yapıydı. İlk olarak Gentry HE bir yapı tasarlamak için kafesleri olmayan idealleri ve halkaları kullandı. Kafes tanımlamaları daha sonra gerçekleştirerek her ideal kafesler ile temsil edildi. Gentry'nin sunduğu SWHE ideal kafes tabanlı yapı aşağıda açıklanmıştır (Acar vd., 2018).

Anahtar Üretimi: R halka, I ideal ve idealin temeli B_I olsun. İdealGen() olarak Gentry tarafından adlandırılan algoritma, temelleri $B_I(I + J = R)$ gibi olan ideal kafesin açık ve gizli anahtarlarının elde edilmesini sağlayan bir algoritmadır. İdealGen(R,B_I) algoritması sonuç olarak (B_J^{sk}, B_J^{pk}) açık ve gizli anahtar çiftini oluşturur. (R, B_I, B_J^{pk}, $Samp()$) açık anahtar ve B_J^{sk} gizli anahtardır. Samp() algoritması kafesin eş kümesinden örnek temel oluşturmak için kullanılır (Acar vd., 2018).

Şifreleme Algoritması: $\vec{r}$ ve $\vec{g}$ rastgele seçilen vektörler, $\vec{m} \in \{0,1^n\}$ açık metin olsun. Şifreleme algoritmasının formülü (Acar vd., 2018);

$$\vec{c} = E(\vec{m}) = \vec{m} + \vec{r} \times B_I + \vec{g} \times B_J^{pk} \tag{4.10}$$

$(\vec{m} + \vec{r} \times B_I)$ gürültü parametresi olarak adlandırılır.

Şifre Çözme Algoritması: B_J^{sk} gizli anahtar temeli, c şifreli metin olsun. Şifre çözme algoritması formülü (Acar vd., 2018);

$$\vec{m} = \vec{c} - B_J^{sk} \times \left\lfloor \left(B_J^{sk}\right)^{-1} \times \vec{c} \right\rceil \ mod\ B_I \tag{4.11}$$

$\lfloor \cdot \rceil$, vektörün katsayıları için en yakın tam sayıyı döndüren en yakın tam sayı işlevidir.

Homomorfik Toplama: $\vec{m}_1$ ve $\vec{m}_2$ açık metinler, Şif() şifreleme işlemi olsun. Homomorfik toplama aşağıdaki şekilde gösterilir (Acar vd., 2018);

$$\vec{c}_1 + \vec{c}_2 = Şif(\vec{m}_1) + Şif(\vec{m}_2) = \vec{m}_1 + \vec{m}_2 + (\vec{r}_1 + \vec{r}_2) \times B_I + (\vec{g}_1 + \vec{g}_2) \times B_j^{pk} \tag{4.12}$$

$\vec{c}_1$ ve $\vec{c}_2$'nin toplama işleminden sonra şifreli metin uzayı içerisinde olduğu görülmektedir. Şifreli metinlerin toplamının şifresini çözmek için;$(\vec{c}_1 + \vec{c}_2) mod\ B_j^{pk}$ hesaplanır. Bu değer $B_j^{pk}/_2$ değerinden küçük olan gürültü değerleri için $\vec{m}_1 + \vec{m}_2 +$ $(\vec{r}_1 + \vec{r}_2) \times B_I$'e eşittir. Bu şekilde gürültünün mod B_I değeri alınarak $m_1 + m_2$ değeri doğru bir şekilde kurtarılır (Acar vd., 2018).

Homomorfik Çarpma: $\vec{e} = \vec{m} + \vec{r} \times B_I$ olsun. Homomorfik çarpma özelliği aşağıdaki şekilde gösterilebilir (Acar vd., 2018);

$$\vec{c}_1 \times \vec{c}_2 = Şif(\vec{m}_1) \times Şif(\vec{m}_2) = \vec{e}_1 \times \vec{e}_2 + (\vec{e}_1 \times \vec{g}_2 + \vec{e}_2 \times \vec{g}_1 + \vec{g}_1 \times \vec{g}_2) \times B_j^{pk} \quad (4.13)$$

$\vec{c}_1$ ve $\vec{c}_2$'nin çarpma işleminden sonra şifreli metin uzayı içerisinde olduğu görülmektedir. Yani, eğer gürültü $|\vec{e}_1 \times \vec{e}_2|$ açık metinlerin çarpımından yeterince küçük $\vec{m}_1 \times \vec{m}_2$ ise doğru bir şekilde şifreli metinlerin çarpımından $\vec{c}_1 \times \vec{c}_2$ kurtarılır (Acar vd., 2018).

4.4.2. Tam Sayı Tabanlı FHE

2010 yılında Marten Van Dijk tarafından Gentry'nin boostrapping yapısını kullanan bir FHE yapısı önerildi. Tam sayı tabanlı olan bu yapı zorluk için En Büyük Ortak Bölen sorununu kullanmaktadır (Yi vd., 2014).

Anahtar Üretimi: n bit uzunluğunda rastgele bir p sayısı üretilir. p gizli anahtardır (Acar vd., 2018).

Şifreleme Algoritması: Rastgele iki adet büyük asal sayı seçilir, p ve a. Rastgele bir adet $p$değerinden küçük bir sayı seçilir, r. $m \in \{0,1\}$ açık metin ve c şifreli metin olacak şekilde (Acar vd., 2018);

$$c = Şif(m) = m + 2r + pq \quad (4.13)$$

Şifre Çözme Algoritması: m açık metin ve c şifreli metin olacak şekilde

$$m = ŞifÇöz(c) = (c \bmod p) \bmod 2 \tag{4.14}$$

Şifre çözme $m + 2r < p/2$ eşitliği sağlandığı sürece geçerlidir. Bu şifreli metin üzerinde gerçekleştirilen homomorfik işlemleri sınırlamaktadır. Squashing ve boostrapping işlemeleri uygulanarak tam homomorfik bir yapı elde edilebilir (Acar vd., 2018).

Homomorfik Toplama: $m_1$ve m_2 açık metinler, Şif() şifreleme işlemi olsun. Homomorfik toplama aşağıdaki şekilde gösterilir;

$$Şif(m_1) + Şif(m_2) = m_1 + 2r_1 + pq_1 + m_2 + 2r_2 + pq_2 = (m_1 + m_2) + 2(r_1 + r_2) + (q_1 + q_2)p \tag{4.15}$$

Şifreli metinler üzerinde yapılan işlemlerin temiz bir sonuç vermesi için $m_1 + 2r_1 + m_2 + 2r_2 < p/2$ eşitliğinin sağlanması gerekmektedir (Acar vd., 2018).

Homomorfik Çarpma: Homomorfik çarpma özelliği aşağıdaki şekilde gösterilebilir (Acar vd., 2018);

$$Şif(m_1) \times Şif(m_2) = (m_1 + 2r_1 + pq_1) \times (m_2 + 2r_2 + pq_2) = m_1m_2 + 2(m_1r_2 + m_2r_1 + 2r_1r_2) + kp \tag{4.16}$$

Şifreli metinler üzerinde yapılan işlemlerin temiz bir sonuç vermesi için $m_1m_2 + 2(m_1r_2 + m_2r_1 + 2r_1r_2) < p/2$ eşitliğinin sağlanması gerekmektedir (Acar vd., 2018).

4.4.3. Hata ile Öğrenme (LWE) Tabanlı FHE

Hata ile öğrenme, kuantum sonrası algoritmalar için bile geçerli bir sürede çözülmesi en zor problemlerden biri olarak kabul edilmektedir. 2011 yılında Brakerski ve Vaikunta-nathan tarafından LWE tabanlı FHE yapısı önerildi. Brakerski ve

Vaikunta-nathan önerdiği yapıdan sonraki FHE yapılarına ikinci nesil FHE yapıları denilmektedir (Tebaa, El Hajji & El Ghazi, 2012).

Anahtar Üretimi: Halkanın bir elemanı hata dağılımından gizli anahtar olarak seçilir, $s \xleftarrow{\$} x$. Daha sonra gizli anahtar vektörü bir D için tanımlanır, $\vec{s} = (1, s, s^2, \dots, s^D)$ (Acar vd., 2018).

Şifreleme Algoritması: Rastgele bir vektör seçilir, $a \xleftarrow{\$} R_q^n$. Gürültü vektörü, $e \xleftarrow{\$} x$. Açık metin m için (Acar vd., 2018);

$$\vec{c} = (c_0, c_1) = (as + te + m, -a) \tag{4.17}$$

Şifre Çözme Algoritması: 4.18'de gösterilen şekilde şifre çözme işlemi gerçekleştirilir.

$$m = \langle \vec{c}, \vec{s} \rangle \pmod{t} \tag{4.18}$$

Şifre çözme işlemi $\langle \vec{c}, \vec{s} \rangle < q/2$ eşitliği sağlandığı sürece geçerlidir.

Homomorfik Toplama: m ve m' açık metinler, $Şif()$ şifreleme işlemi olsun. Homomorfik toplama aşağıdaki şekilde gösterilir (Acar vd., 2018);

$$Şif(m) + Şif(m') = (c_0 + c_0', c_1 + c_1') = ((a + a')s + t(e + e') + (m + m'), -(a + a')) \tag{4.19}$$

Homomorfik toplama özelliğinin çalışması için gürültünün küçük olması gerekmektedir (Acar vd., 2018).

Homomorfik Çarpma: Homomorfik çarpma özelliği aşağıdaki şekilde gösterilebilir.

$$Şif(m) \times Şif(m') = (c_0 c_0', c_1 c_1') = (-a's^2 + (c_0' a + c_0 a')s + t(2ee' + em' + e'm) + mm') \tag{4.20}$$

Teorik olarak, ancak şifreli metne yeni bir ekleme yapılarak doğru bir şekilde şifrelenebilir. Buda yeni bir maliyet doğuracaktır (Acar vd., 2018).

Brakerski ve Vaikunta-nathan Gentry'nin boostrapping ve squashing yöntemlerini kullanarak yeni bir FHE yapısı önerdi. Bu çalışmalarında SWHE şemalarının, gizli anahtarın doğrusal fonksiyonlara göre dairesel güvenli olduğunu gösterdi (Acar vd., 2018).

4.4.4. NTRU Benzeri FHE

NTRU şifreleme yapısı kafes problemlerine dayanan en eski çalışmalardan biridir. 1998 yılında Hoffstein, Pipher ve Silvermanin tarafından NTRU şifreleme şeması gerçekleştirildi. 2012 yılında López-Alt vd. tarafından NTRU benzeri bir FHE yapısı önerildi. NTRU çok anahtarlı yapıya sahip bir şifreleme yapısıdır. RSA gibi şifreleme yapıları ile karşılaştırıldığında hem donanım hem de yazılım uygulamaların verimliliğini önemli derece arttırmaktadır. López-Alt vd. tarafından önerilen NTRU benzeri FHE yapısı aşağıdaki şekildedir (Sen, 2013);

Anahtar Üretimi: Seçilen örneklenmiş polinom f' ve x dağılımından g (Gauss dağılımı), $f = 2f' + 1$, $f \equiv 1\ (mod\ 2)$ ve f ters alınabilirdir. Gizli anahtar $ga = f \in R$ ve açık anahtar $aa = h = 2gf^{-1} \in R_q$'dır (Acar vd., 2018).

Şifreleme Algoritması: s ve e x dağılımından seçilen örnekler ve m açık metin ve $c \in R_q$ şifreli metin olacak şekilde (Acar vd., 2018);

$$c = Şif(m) = hs + 2e + m \tag{4.21}$$

Şifre Çözme Algoritması: $fc \in R_q$ olacak şekilde (Acar vd., 2018);

$$m = ŞifÇöz(c) = fc\ (mod\ 2) \tag{4.22}$$

Homomorfik Toplama: m_i açık metin, $h_i = 2g_i f_i^{-1}$ açık anahtar, $c_i = h_i s_i + 2e_i + m_i$ şifreli metin (Acar vd., 2018).

$$f_1 f_2(c_1 + c_2) = 2(f_1 f_2 e_1 + f_1 f_2 e_2 + f_2 g_1 s_1 + f_1 g_2 s_2) + f_1 f_2(m_1 + m_2) = 2e_{toplam} + f_1 f_2(m_1 + m_2) \tag{4.23}$$

Homomorfik Çarpma: Homomorfik çarpma özelliği aşağıdaki şekilde gösterilebilir (Acar vd., 2018);

$$f_1 f_2(c_1 c_2) = 2\big(2g_1 g_2 s_1 s_2 + g_1 s_1 f_2(2e_2 m_2) + g_2 s_2 f_1(2e_1 + m_1) + f_1 f_2(e_1 m_2 + e_2 m_1 + 2e_1 e_2)\big) + f_1 f_2(m_1 m_2) = 2e_{çarpım} + f_1 f_2(m_1 m_2) \quad (4.24)$$

Gürültü parametreleri e_{toplam} ve $e_{çarpma}$ değerleri $p/2$ değerinden küçük olduğu sürece şifre çözme işlemi geçerlidir (Acar vd., 2018).

BÖLÜM 5

HOMOMORFİK ŞİFRELEME UYGULAMALARI

5.1. Giriş

Tezin bu bölümünde HE yönteminin potansiyel uygulama alanları hakkında bilgi vererek HE yönteminin önemine değinilmiştir.

5.2. Genom ve HE

Günümüzde insanların genom haritaları çıkarılma çalışmaları hızlı bir şekilde sürmektedir. Yakın bir gelecekte dünya üzerindeki insanların birçoğunun genom haritalarına sahip olacağız. Biyoloji, tıp ve insanlığın varoluşu konuları hakkında bu bilgilerin bize yardımcı olmaları beklenmektedir. Genom haritaları bilgisi laboratuarlarda ve tıp merkezlerinde tutulmaktadır. Fakat bu verilerin saklanmasında ve paylaşılmasında insanların güvenliği ve gizliliği açısından sıkıntılar doğmaktadır. DNA ve RNA bir insanın parmak izi gibi özel bir tanımıdır. Bu bilgilerin güvenliğinin sağlanmadan paylaşılması büyük bir sorun teşkil etmektedir (Archer vd., 2017).

Genom verilerin korunması National Institutes of Health tarafından sağlanmaktadır. Bu verilere güvenli erişim için devlet kontrolü altında veya güvenilir ortaklar tarafından erişim sağlanmaktadır. Bu verilerin hızlı bir şekilde artmasından dolayı bulut tabanlı yeni çalışmalar önerildi. Fakat şuan için bu verilerin bulut ortamına aktarılması güvenli değildir (Archer vd., 2017).

Genom verilerinin bazıları veriler üzerinde işlemlerin yapılarak belirli bir sonuç elde etmeye dayanmaktadır. HE yöntemi ile buluta yüklenecek genom verilerinin bazıları üzerinde güvenli bir şekilde işlem yapılabilirliği sağlanabilir (Archer vd., 2017).

5.3. Ulusal Güvenlik ve HE

Hükümete bağlı bulunan bina, şebeke, jeneratör vb. gibi yapıların birbiri ile bağlantılığı olduğu akıllı bir şebeke ağı oluştuğunda, bu yapılar belirli bir veri üretecektir. Bu verilerin çeşitli hükümet organları ile paylaşılarak gerekli analizlerin ve incelemelerin yapılması gerekmektedir. Bu işlemlerin güvenli bir şekilde yapılabilmesi için HE yapısından faydalanarak gerekli hesaplamalar ve incelemeler yapılabilir. Bu tür bilgiler ile kötü amaçlı yazılımlar, anormallikler ve izinsiz girişlerin tespiti yapılarak güvenlik sağlanabilir (Archer vd., 2017).

Böyle bir yapı çok fazla alandan çok fazla bilgi içerdiği için bu bilgileri korumak ve kötü kişiler tarafından ele geçirilmediğini sağmak çok önemlidir. Örnek olarak, şebeke ve güç dağılımı bilgilerinin kötü kişiler tarafından ele geçirildiğinde, bu alanlara saldırılar olabilir (Archer vd., 2017).

Herhangi başka bir uygulama ile bu güvenliği sağlamak, uygulamada kullanılan yöntemlerin belirgin olamaması, algoritmaların bilinmemesi ve protokollerin güvenilirliğinin test edilememesi durumlarından dolayı kesin değildir. Bu nedenle kritik alanlarda HE yöntemi kullanmak çok daha iyi olabilmektedir(Archer vd., 2017).

5.3. Eğitim ve HE

ABD'de her geçen yıl 1 milyondan fazla öğrenci lise zamanı okulu bırakmaktadır (2017 verilerine göre). Liseye başlayan öğrencilerin çeyreği okulunu zamanında bitirememektedir. Bu problemi en aza indirgemek için; liseyi bitirememe ve liseyi zamanında tamamlayamama grubunda olabilecek öğrencileri liseye başlamadan önce tespit edip gerekli önlemler alınabilir. Böyle bir tespitin yapılabilmesi için hükümete bağlı okul, hastane, polise vb. kurumlardaki bilgilerin birleştirerek analiz edilmesi gerekmektedir.Fakat bu kurumların hepsinin kendi bilgilerini koruma ve güvenliğin sağlamakla yükümlüdür (Archer vd., 2017).

Bütün bu bilgilerin tek yerde tutulması büyük bir risk oluşturmaktadır. Bu denli büyük bir bilgi kötü kişiler tarafından ele geçirmeye çalışılabilir ve sürekli saldırı altında olabilir (Archer vd., 2017).

HE yapısından faydalanarak bu bilgilerin herhangi bir yere taşınmasına gerek kalmadan hesaplanabilir hale getirilebilir. Bu şekilde güvenlik en üst düzeyde tutulmuş olacaktır (Archer vd., 2017).

5.4. Sağlık ve HE

Sağlık hizmetlerinde kişilerin hassas bilgileri bulunmaktadır. Bu bilgilerin güvenliğinin iyi sağlanması kötü kişiler tarafından ele geçirilmemesi çok önemlidir. Fakat bir yandan bu bilgilerin sürekli olarak işlenmesi gerekmektedir. Risk ve fayda arasındaki bu bilgilerin doğru bir şekilde korunmadığında bilgi kaybının yanında büyük maliyetlere sebep olduğu görülmektedir (Archer vd., 2017).

HE yapısı sağlık hizmetlerindeki bilgilerin risk ve fayda dengesini sağlamada rol oynayabilir. Fatura, rapor oluşturma gibi yapılacak işlemlerin yapılması sırasında bilgilerin gizli kalmasını sağlayıp sadece sonucun elde edilmesini sağlayabilir. Bu şekilde verilerin güvenliği sağlanarak, büyük maliyetlere sebebiyet verecek olaylardan korunmasını sağlar (Archer vd., 2017).

5.5. Bulut Bilişim ve HE

Bulut Bilişim, günümüzde popüler bir şekilde kullanılan internet üzerinde verilerin depolanabileceği bir teknolojidir. Bulut bilişim, ulaşılabilirliğin kolay olması ve maliyetinin düşük olmasından dolayı çok tercih edilmektedir. Aynı zamanda kişisel verilerin bu teknoloji üzerinde tutulmasından dolayı, saldırganlar için potansiyel bir saldırma yeridir. Saldırıya uğrayan bir bulutun, kişisel verileri nasıl koruyacağına dair sorunlar gündeme gelmektedir. Bu sorunu çözebilmek için bulut üzerinde tutulan verilerin şifreli bir şekilde saklanması gerekmektedir. Saldırgana kullanıcı verilerine ulaşabile şifreli oldukları için herhangi bir şey elde edemez. Fakat verilerin şifreli bir şekilde tutulması kullanıcıların verilerine ulaşmak istemesi veya üzerinde değişiklikler yapmak istemesi halinde şifreleme ve şifre çözme işlemleri gerçekleştirmek gerekmektedir. Bu işlemler ek maliyetler getirecektir (Çalık vd., 2015).

HE yapısı bulut bilişiminde kullanılabilir. HE yapısı sayesinde şifrelenmiş olarak saklanan veriler üzerinde kullanıcılar gerekli işlemleri gerçekleştirerek maliyet düşürülebilir. Veriler şifreli olacağı için saldırganlar tarafından da korunaklı bir yapı haline gelecektir (Çalık vd., 2015).

5.6. Mobil Ajan ve HE

Mobil ajanlar, kullanıcıların önceden belirlenmiş olan görevleri kullanıcıların müdahalesi gerekmeden başka bilgisayarlar üzerinde gerçekleştiren programlardır (Jansen & Karygiannis, 1998). Mobil ajanların çalıştığı diğer bilgisayarların kötü niyetli kişiler tarafınca kontrol edilebilmesinden dolayı güvenlik açığı ortaya çıkmaktadır. Bundan dolayı mobil ajanları kötü niyetli kişiler tarafından korumak ve işlemlerini gerçekleştirmek için HE yapısı kullanılabilir. Verileri, mobil ajanlar tarafından başka bilgisayarlarda kullanılmadan önce şifrelemek ve sonrasında şifreli veriler üzerinde hesaplama yapmak için HE yapısından faydalanılabilir (Sen, 2013).

5.7. Oylama ve HE

Elektronik ortamda gizli oylama sistemlerinde, oylar üzerinde işlemlerin yapılması gerekmektedir. Yapılacak olan bu işlemler sırasında, oy veren bireylerin kullandıkları oy hakkındaki bilgilerin gizlilik nedeni ile ortaya çıkarılmaması gerekmektedir. Şifreli olarak saklanan oylar üzerinde işlem yapabilmek için HE yapısı kullanılabilir. Bu sayede oylar üzerinde analizler yapılırken, oyların gizliliği sürdürülebilir (Sen, 2013).

5.8. Avuç ve Parmak İzi ve HE

Avuç ve parmak izi tanıma teknolojileri özellikle son yıllarda daha fazla kullanılmaya başlamıştır. Avuç ve parmak uçlarındaki deri üzerindeki dalgalanmaların karakteristik bir yapıya sahip olmasında dolayı, bu veriler ile tanımlama işlemleri gerçekleştirilmektedir (Sönmez, Özbek & Özbek, 2007). Bu uygulamaların sahip olduğu kişisel avuç ve parmak izi verilerinin şifreli bir şekilde tutularak ve gerektiğinde bu işlemlerin şifreli bir şekilde yapılması güvenlik için önemlidir. HE yapısı ile bu uygulamalardaki güvenlik arttırılabilir (Sen, 2013).

5.9. Finans ve HE

Finansal sektörde müşteriler tarafından şirketlere yapılacak yatırımlar, şirket hakkındaki bilgiler doğrultusunda yapılmaktadır. Bu bilgiler, şirketin performansı, envanter durumu gibi bilgilerden oluşmaktadır. Bu bilgiler üzerinde hesaplamalar

yapılarak şirket hakkında gerekli bilgiye sahip olan müşteri yatırım yapıp yapmayacağına karar verir. Şirketler bu bilgileri avantajlarını koruyabilmek için gizli tutmak isteyebilirler (Tebaa vd., 2012).

Şirketler HE yapısı kullanılarak bu bilgileri açık bir şekilde yayınlamadan müşteriler tarafından kullanılabilir bir yapı oluşturabilir. Bu sayede şirketler bilgilerini açık bir şekilde paylaşmamış ve müşteriler şirket bilgileri hakkında yapmak istedikleri hesaplamaları yapabilirler (Tebaa vd., 2012).

5.10. Reklam ve HE

Reklam, satıcının sattığı ürün veya hizmetin müşteriler tarafından bilinmesini sağlayan bir araçtır. Reklam veren satıcıların asıl ulaşmak istedikleri kişiler potansiyel alıcılardır. Potansiyel alıcıların kimler olduğunu belirlemek için, insanların kullandıkları bilgisayar, telefon gibi iletişim içerisinde olan elektronik eşyalar kullanılır. Bu eşyaları kullanan insanların; aradıkları kelimeler, ilgi alanları, e-postalar, konum bilgileri gibi bilgilerini toplayarak, o kişi hakkında bilgi sahibi olunabilir. Toplanan bu bilgiler sayesinde reklam yapan satıcıların sattığı ürünün veya hizmetin potansiyel alıcısı olduğu tahmin edilen kişilere sunulması. Bu sayede satıcıların müşterileri ile kolay bir şekilde buluşması sağlanabilir (Tebaa vd., 2012).

HE yapısı sayesinde kişilerden toplanan kişisel bilgilerin açık olarak saklanması gerekmemektedir. Şifreli olarak saklanan bilgilerin, gerektiğinde şifreli olarak işlem yapılarak sonuçları üzerinden potansiyel alıcılar belirlenebilir (Tebaa vd., 2012).

5.11. Kablosuz Algılayıcı Ağlarda Güvenli Veri Kümeleme ve HE

Veri kümeleme, ağ içerisinde bulunan algılayıcıların veri miktarlarını düşürerek enerji tasarrufu yapmasını sağlayan bir yapıdır. Güvenlik sorunları ile karşılaşılmaması için veri gönderecek tarafın veriyi şifreleyerek baz istasyonuna göndermesi gerekmektedir. Fakat verinin gideceği yol üzerindeki algılayıcılar veriyi açık metin haline getirerek veri kümele işlemi yapması gerekmektedir. Güvenlik ve veri kümeleme birbirlerine zıt bir şekilde çalışmaktadır (Özdemir, 2008).

Veriyi baz istasyonuna gönderen taraf veriyi simetrik anahtar şifreleme ile göndermektedir. Ağ üzerindeki algılayıcılar şifre çözme işlemi ile veriyi çözer, veri

kümeleme yapar ve sonrasında tekrar şifreleme işlemini gerçekleştirerek veriyi gönderir. Veri bu işlemler sırasında gizliliğini yitirmektedir (Özdemir, 2008).

Hem veri gizliliğini sağlamak hem de veri kümele işlemini gerçekleştirmek için HE yapısından faydalanılabilir. HE yapısı sayesinde veriyi baz istasyonuna gönderen taraf veriyi şifreleyerek gönderir. Ağ üzerindeki algılayıcılar şifre çözme işlemini gerçekleştirmeden veri kümele işlemini yapar ve veriyi gönderir. Bu sayede şifreli veri yol boyunca açılmadı için veri gizliliği sağlanmış ve veri kümele işlemi gerçekleştirilmiş olur (Özdemir, 2008).

5.12. Internet of Things (IoT) ve HE

IoT, etrafımıza etkide bulunan veya analiz etmemizi sağlayan cihazları etkileşim içinde olmalarını sağlayan bir iletişim ağıdır. Bu iletişim ağı sayesinde üretim süreçlerini, hasta takipleri, geri dönüşüm süreçleri, akıllı binalar gibi alanlarda kontrol sağlanabilir ve analiz yapılabilir (Gökrem & Bozuklu, 2016).

Standart şifreleme yöntemlerinden faydalanarak IoT oluşturulduğunda iki tane problem ortaya çıkmaktadır. Eğer veriler şifrelenmemiş olarak depolanırsa, kötü niyetli kişiler tarafından ele geçirilebilir. Eğer veriler şifrelenmiş olarak depolanırsa, sağlayıcılar üzerinde çalışması için şifrenin çözülmesi gerekir (Chellappan & Sivalingam, 2016).

HE sayesinde veriler şifrelenmiş şekilde depolanarak, şifre çözme işlemine gerek kalmadan sağlayıcılar üzerinde işlem yapılabilir. RSA algoritması, ElGamal algoritması, Paillier algoritması bu amaç ile kullanılabilir. IoT için Homomorfik Şifreleme çok önemlidir. Çünkü bir yandan veri gizliliğini koruyarak, bir yandan da sağlayıcılar ve düğüm noktalarında hızlı bir şekilde şifre çözme ve şifreleme işlemlerine gerek duymadan gerçekleştirilebilir. Düğümlerde toplama ve çarpma işlemleri gerçekleştirilerek işlem ve depolama maliyeti azaltılabilir. Bu sayede düşük güç tüketimi gerçekleşmiş olur (Chellappan & Sivalingam, 2016).

5.13. Veri Tabanı Güvenliği ve HE

Veri tabanı düzenli şekilde verilerin saklandığı bilgi topluluğudur. Veri tabanındaki verilere istenildiği zaman ulaşılabilir ve kullanılabilir (Çiftçi, 2017, s.165). Bu veriler veri tabanında güvenlik için şifreli bir şekilde bulunmaktadır. Fakat veriler

üzerinde gerçekleştirilecek işlemlerin yapılması sırasında bu verilerin şifrelerinin çözülmesi ve açık metin üzerinde işlemlerin gerçekleşmesi gerekmektedir. Bu nedenle, veri tabanındaki veriler güvenlik riskleri ile karşı karşıyadır (Gahi, Guennoun & El-Khatib, 2015).

HE yapısı veriler üzerinde şifre çözme işlemini gerçekleştirmeden işlemler yapılabilmesine olanak sağlamaktadır. Bu sayede veri tabanında saklanan veriler üzerinde işlemler güvenli bir şekilde gerçekleşebilir (Gahi vd., 2015).

5.14. Ağ Kontrol Sistemleri ve HE

Ağ kontrol sistemleri siber güvenliğin en önemli konularından biridir. Su, ulaşım ve elektrik ağları gibi endüstriyel ve kritik altyapılara uygulanır. Kötü niyetli kişiler bu ağlara karşı saldırılarda bulunmaktadır. İletişim kanalları üzerinden gönderilen sinyallerin güvenliği sağlanmalıdır (Kogiso & Fujita, 2015).

Kontrol cihazının parametrelerini ve kontrol cihazı içerisindeki sinyalleri şifrelemek güvenlik açısından önemlidir. Kontrol cihazına gelene sinyal şifreli bir şekilde gelerek şifre çözme işlemi gerçekleştirilir ve gerekli işlemlerin yapılması sonucunda tekrar şifrelenerek şifreli olarak kontrol cihazından çıkış gerçekleştirir. HE yapısı kontrol cihazlarında uygulandığında şifreli olarak gelen sinyal şifre çözme işlemi gerçekleşmeden gerekli işlemler üzerinde gerçekleştirilerek şifreli olarak çıkış gerçekleştirir. Bu sayede ağ kontrol cihazları üzerinde güvenlik arttırılmış olacaktır (Kogiso & Fujita, 2015).

5.15. Kapalı Teklif Açık Arttırma ve HE

Açık arttırma bir malın, menkulün, eşyanın vb. ürünlerin alıcı ve satıcıların bir arada bulunarak fiyat tekliflerinin bildirilmesi ile en yüksek fiyata sahip alıcının ürünü almasıdır. Açık arttırmalar açık teklif veya kapalı teklif şeklinde ikiye ayrılmaktadır. Açık teklif şeklinde gerçekleşen açık arttırmalarda, alıcılar fiyat tekliflerini açık bir şekilde gerçekleştirirler. Kapalı teklif şeklinde gerçekleşen açık arttırmalarda, alıcılar fiyat tekliflerini kapalı bir şekilde gerçekleştirirler (Oran, 2008).

Kapalı teklif açık arttırmalarda, alıcıların verdikleri teklifler gizli olmalıdır. Bu tekliflerin açığa çıkması açık arttırmayı olumsuz yönde etkilemektedir. Bir yandan bu

tekliflerin karşılaştırma işlemleri gerçekleştirilmesi gerekmektedir. Burada üçüncü şahıslara olan güven esas alınmaktadır (Suzuki & Yokoo, 2003).

HE yapısı kapalı teklif açık arttırmalarda uygulanabilir. Bu sayede alıcılar tarafından verilen teklifler şifreli bir şekilde saklanır. Karşılaştırmaların yapılması için şifreli olarak saklanan veriler şifre çözme işlemine gerek olmadan gerekli işlemler üzerinde gerçekleştirilebilir. Bunun sonucunda, teklifler üçüncü bir şahıssa güven olmayı gerektirmeden açık arttırma gerçekleşebilir (Suzuki & Yokoo, 2003).

BÖLÜM 6

SONUÇ

Bu tez çalışmasında; Homomorfik Şifreleme yöntemi ve kullanım alanları incelenmiştir.

Homomorfik Şifreleme yöntemi veriyi bozmadan, şifreli veriler üzerinde matematik işlemler yapmaya olanak sağlayan bir yöntemdir. Homomorfik Şifreleme yönteminin temellerini oluşturan RSA algoritması, ElGamal algoritması ve Paillier algoritmaları detaylı bir şekilde incelenmiştir. RSA algoritması çarpma işlemine göre, ElGamal algoritması ve Paillier algoritması toplama işlemine göre homomorfik özellik gösterdikleri örnekler ile incelenmiştir. Bu algoritmalar tek bir işlem tipinde homomorfik özellik gösterdikleri için Kısmi-Homomorfik Şifreleme olarak adlandırılmıştır. Kısmi-Homomorfik Şifreleme yöntemleri ya sadece toplama ya da sadece çarpma işlemlerine olanak sağladığı için yeterli olmamıştır.

Kısmi-Homomorfik Şifreleme yöntemlerinin yeterli olmamasından dolayı Homomorfik Benzeri Şifreleme yöntemleri geliştirilmiştir. Bu yöntemler Kısmi-Homomorfik Şifreleme yöntemlerindeki tek bir işlem tipine olanak sağlamasını gidermek için hem toplama hem de çarpma işlemlerini bir arada yapmayı hedeflemiştir. Fakat Homomorfik Benzeri Şifreleme yöntemleri istenilen yönde ilerle göstermiş olmasına rağmen yeterli olamamıştır. Homomorfik Benzeri Şifreleme yöntemleri toplama işlemine sınırsız sayıda yapılmasına olanak sağlarken çarpma işlemine tek bir sefer yapılmasına olanak sağlamaktadır. Homomorfik Benzeri Şifreleme yöntemi tam olarak istenilen yapıyı sağlamamış olmasına rağmen büyük bir gelişme kaydedilmesini sağlamıştır. BGN şifreleme yöntemi Homomorfik Benzeri Şifreleme yöntemine örnek bir algoritmadır.

Tam-Homomorfik Şifreleme yöntemi Homomorfik Benzeri Şifreleme yönteminin geliştirilmiş hali olarak düşünülebilir. 2009 yılında Craig Gentry tarafından gündeme gelen Tam-Homomorfik Şifreleme yöntemi, Homomorfik Benzeri Şifreleme

yönteminin sağlayamadığı sınırsız sayıda çarpma işlemine olanak saylayarak, hem toplama hem de çarpma işlemlerini sınırsız sayıda yapılmasına olanak sağlamayı hedeflemiştir. Gentry'nin sunduğu Tam-Homomorfik Şifreleme yöntemi maliyet gibi problemler içermekteydi. Fakat Tam-Homomorfik Şifreleme yapısının nasıl olabileceği hakkında bir yol göstermişti. Tam-Homomorfik Şifreleme yöntemi maliyet problemlerinin giderilebilmesi için farklı tipte yöntemler ve zorluklar kullanılarak geliştirilmiş ve geliştirilmeye devam etmektedir.

Homomorfik Şifreleme yöntemleri birçok alanda uygulanabilir. Veriler üzerinde işlemlerin gerçekleştirilebilmesi ve aynı zamanda verilerin gizliliğinin korunması istenilen alanlarda kullanılabilir. Gen, sağlık, güvenlik, ulusal güvenlik, eğitim, finans, mobil ajan, bulutum bilişim, oylama sistemi, avuç ve parmak izi, reklam, kablosuz ağlar, Internet of Things, veri tabanı güvenliği Homomorfik Şifreleme yönteminin kullanılabileceği alanlara örnek olarak verilebilir. Homomorfik Şifreleme sayesinde veriler üzerinde rahatça hesaplamalar yapılarak olasılıklar, analizler hesaplanabilecek ve verilerin gizliliği hiçbir şekilde bozulmamış olacaktır.

KAYNAKLAR

Acar, A., Aksu, H., Uluagac, A. S., Conti, M. (2018). *A survey on homomorphic encryption schemes: Theory and implementation*.

Akkaş, S., Hacısalihoğlu, H. H., Özel, Z., Sabuncuoğlu, A. (1998). *Soyut Matematik (3. Baskı). Ankara: Gazi Üniversitesi.*

Archer, D., Chen, L., Cheon, J. H., Gilad-Bachrach, R., Hallman, R. A., Huang, Z., Jiang, x., Kumaresan, R., Malin, B. A., Sofia, H., Song, Y., Wang, S. (2017). *Applications of homomorphic encryption*.

Asar, A. O., Arıkan, A., Arıkan A. (2012). *Cebir (2. Baskı). Ankara: Sincan Matbaası.*

Boneh, D., Goh, E. J., Nissim, K. (2005). *Evaluating 2-DNF formulas on ciphertexts. In Theory of Cryptography Conference (s. 325-341). Springer, Berlin, Heidelberg.*

Canberk, G. & Sağıroğlu, Ş. (2006). *Bilgi, bilgi güvenliği ve süreçleri üzerine bir inceleme. Politeknik Dergisi*, *9*(3), 165-174.

Chellappan, V., & Sivalingam, K. M. (2016). *Security and privacy in the Internet of Things. In Internet of Things (s. 183-200). Morgan Kaufmann.*

Çalık, E., Erdem, H. A., Aydın, M. A. (2015) *"Bulut Bilişim Güvenliği için Homomorfik Şifreleme"*.

Çakır, C. (2016). *İktisat tarihi çalışmalarına bir katkı:"Avrupa devletlerinin nüfus ve arazi istatistiği". Dîvân: Disiplinlerarası Çalışmalar Dergisi*, (4), 191-206.

Çelik, B., Cangül, İ. N., Çelik, N., Bizim, O., Öztürk, M. (2010). *Temel Matematik (5. Baskı). Bursa: Dora.*

Çiftçi, N. (2017). *Veritabanı. Pegem Atıf İndeksi, s. 165-211.*

Demir, M. K. (2014). *Esnek Kümeler ve Esnek Cebirsel Yapılar (Master's thesis, Fen Bilimleri Enstitüsü).*

ElGamal, T. (1984). *A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms.*

Ellison, C. (2001). *Cryptography timeline.*

Forouzan, B. (2008). *Cryptography and Network Security: McGraw-Hill International Edition.*

Gahi, Y., Guennoun, M., & El-Khatib, K. (2015). *A secure database system using homomorphic encryption schemes.*

Gentry, C. (2009). Fully *homomorphic encryption using ideal lattices*.

Gökrem, L., & Bozuklu, M. (2016). *Nesnelerin interneti: Yapılan çalışmalar ve ülkemizdeki mevcut durum. Gaziosmanpaşa Bilimsel Araştırma Dergisi, (13), 47-68.*

Ishai, Y., & Paskin, A. (2007). *Evaluating branching programs on encrypted data. In Theory of Cryptography Conference (s. 575-594). Springer, Berlin, Heidelberg.*

Jansen, W., & Karygiannis, T. (1998). *Mobile agent security.*

Kodaz, H. & Botsalı, F. M. (2010). *Simetrik ve asimetrik şifreleme algoritmalarının karşılaştırılması.*

Kogiso, K., & Fujita, T. (2015). *Cyber-security enhancement of networked control systems using homomorphic encryption. In 2015 54th IEEE Conference on Decision and Control (CDC) (s. 6836-6843).*

Lai, X., Zhou, J., Li, H. (2011). *Information Security: 14th International Conference, ISC 2011. Springer Science & Business Media.*

Ogburn, M., Turner, C., Dahal, P. (2013). *Homomorphic encryption. Procedia Computer Science.*

Oran, J. S. (2008). *Açık Arttırma; Tarihçe, Genel Bilgiler ve Bir Araştırma.*

Özdemir, S. (2008). *Kablosuz Algılayıcı Ağlarında Homomorfik Şifremele ile Güvenli Veri Kümeleme. Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 23(2).*

Paillier, P. (1999). *Public-Key Cryptosystems Based on Composite Degree Residuosity Classes.*

Parmar, P. V., Padhar, S. B., Patel, S. N., Bhatt, N. I., Jhaveri, R. H. (2014). *Survey of various homomorphic encryption algorithms and schemes. International Journal of Computer Applications*.

Pettersen, N. (2016). *Applications of Paillier s Cryptosystem*.

Rivest, R., Shamir, A., Adleman, L. (1978). *A method for obtaining digital signatures and public-key cryptosystems.*

Sakallı, M. T. (2006). *Modern şifreleme yöntemlerinin gücünün incelenmesi.*

Sander, T., Young, A., Yung, M. (1999). *Non-interactive cryptocomputing for nc/sup 1. In 40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039) (s. 554-566).*

Sen, J. (2013). *Homomorphic encryption: Theory & applications*.

Sönmez, E. B., Özbek, N. Ö., & Özbek, Ö. (2007). *Avuç izi ve parmak izine dayalı bir biyometrik tanıma sistemi. Akademik Bilişim'07-IX. Akademik Bilişim Konferansı, s. 577-582.*

Suzuki, K., & Yokoo, M. (2003). *Secure generalized vickrey auction using homomorphic encryption. In International Conference on Financial Cryptography (s. 239-249). Springer, Berlin, Heidelberg.*

Tebaa, M., El Hajji, S., El Ghazi, A. (2012). *Homomorphic encryption applied to the cloud computing security, In Proceedings of the World Congress on Engineering.*

Tuncay, G. (2015). *Kriptografik ve uygulama özellikleri iyi olan ikili matrislerin üretilmesi için yeni bir arama yöntemi.*

Ural, N., & Örenç, Ö. (2018). *Şifreleme ve Şifre Çözme Yöntemleri: Uygulamalı (2. Baskı). İstanbul: Pusula.*

Yao, A. C. (1982). Protocols for secure computations. In 23rd annual symposium on foundations of computer science (s. 160-164).

Yerlikaya, T., Buluş, E., Buluş, N. (2006). *Asimetrik Şifreleme Algoritmalarında Anahtar Değişim Sistemleri.*

Yi, X., Paulet, R., Bertino, E. (2014). *Homomorphic encryption and applications (3. Baskı). Heidelberg: Springer.*

ÖZGEÇMİŞ

1990 yılında İzmir'de doğdum. İlk, orta ve lise öğretimini Edirne'de tamamladıktan sonra, 2010 yılında Kocaeli Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği bölümünü kazandım. 2014 yılında mezun olduktan sonra 2016 yılında Trakya Üniversitesi Fen Bilimleri Enstitüsü Hesaplamalı Bilimler bölümünde tezli yüksek lisansa ve Trakya Üniversitesi Bilgi İşlem Daire Başkanlığında çalışmaya başladım. 2018 yılında Trakya Üniversitesi Fen Bilimleri Enstitüsü Hesaplamalı Bilimler bölümünden Trakya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği bölümüne geçiş yaptım. İş hayatıma Trakya Üniversitesi Bilgi İşlem Daire Başkanlığında ve öğretim hayatıma Trakya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği bölümünde devam etmekteyim.