GRUP ETKİSİNE DAYALI ANAHTAR ANLAŞMASI PROTOKOLLERİNE GENEL BİR BAKIŞ

Ahmet ÇÖP

Yüksek Lisans Tezi

Matematik Anabilim Dalı

Dr. Öğr. Üyesi Abdullah ÇAĞMAN

AĞRI-2021

T.C.

AĞRI İBRAHİM ÇEÇEN ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

MATEMATİK ANABİLİM DALI

Ahmet ÇÖP

GRUP ETKİSİNE DAYALI ANAHTAR ANLAŞMASI PROTOKOLLERİNE GENEL BİR BAKIŞ

YÜKSEK LİSANS TEZİ

TEZ YÖNETİCİSİ

Dr. Öğr. Üyesi Abdullah ÇAĞMAN

AĞRI-2021

ÖZET

YÜKSEK LİSANS TEZİ

GRUP ETKİSİNE DAYALI ANAHTAR ANLAŞMASI PROTOKOLLERİNE GENEL BİR BAKIŞ

Tez Danışmanı: Dr. Öğr. Üyesi Abdullah ÇAĞMAN

2021, 45 sayfa

Jüri: Dr. Öğr. Üyesi Sait TAŞ

Dr. Öğr. Üyesi Abdullah ÇAĞMAN

Dr. Öğr. Üyesi Kadirhan POLAT

Bu tezde Anahtar anlaşması protokollerinden bahsederek, grup etkisine dayalı anahtar anlaşması protokolleri incelenmiştir. Tezin giriş bölümünde, konunun tarihçesi ve ilk çalışmanın ortaya çıkışı, ikinci bölümde, konumuzla alakalı temel tanım ve teoremler, üçüncü bölümde, etki ve anahtar anlaşması protokolleri, dördüncü bölümde, Yarı-Grup ve Grup Etkisi Üzerine Yapılan Anahtar Anlaşması Protokolleri incelenerek son bölümde sonuçlara yer verilmiştir.

2021, 45 sayfa

Anahtar Kelimeler: Anahtar Anlaşması Protokolü, Grup Etkisi, Yarı Grup.

ABSTRACT

MASTER'S THESİS

AN OVERVIEW OF KEY AGREEMENT PROTOCOLS BASED ON GROUP ACTION

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

2021, 45 Pages

Jury : Assist. Prof. Dr. Sait TAŞ

Assist. Prof. Dr. Abdullah ÇAĞMAN

Assist. Prof. Dr. Kadirhan POLAT

In this thesis, by mentioning key agreement protocols, key agreement protocols based on group effect have been examined. In the introduction part of the thesis, the history of the subject and the emergence of the first study, in the second part, the basic definitions and theorems related to our topic, in the third part, the impact and key agreement protocols, in the fourth part, the Key Agreement Protocols on Semi-Group and Group Effect are examined and the conclusions are included in the last part.

2021, 45 Pages

Key Words: Group Action, Key Agreement Protocol, Semi-Group.

TEŞEKKÜR

Bu tez çalışması Ağrı İbrahim Çeçen Üniversitesi Lisansüstü Eğitim Enstitüsü Matematik Anabilim Dalında hazırlanmıştır.

Yüksek lisans eğitimim boyunca, benden bilgi ve desteğini esirgemeyen, çalışmalarımın tamamlanabilmesi için her türlü şartı sağlayan kıymetli danışman hocam Sayın Dr. Öğr. Üyesi Abdullah ÇAĞMAN’ a teşekkürlerimi sunarım.

Öğrenim hayatım boyunca kendilerinden görmüş olduğum maddi ve manevi destek den dolayı aileme ve her zaman her türlü desteği ile yanımda olan sevgili eşim Zehra ÇELİK ÇÖP' e teşekkürlerimi sunarım.

24/05/2021

Ahmet ÇÖP

İÇİNDEKİLER

SİMGELER ve KISALTMALAR DİZİNİ

$<$	Küçüktür
$>$	Büyüktür
$\leq$	Küçük veya Eşittir
$\geq$	Büyük veya Eşittir
$\subset$	Alt Küme
$\subseteq$	Alt Kümesi veya Eşit
$\subsetneq$	Alt Küme ve Eşit Değil
$\cong$	Yaklaşık Olarak Eşit
$\ni$	Eleman Olarak Kapsar
$\in$	Elemanıdır
$\notin$	Elemanı Değildir
$\mathbb{N}$	Doğal Sayılar Kümesi
$\mathbb{R}$	Reel Sayılar Kümesi
max	Maksimum
$\sum$	Toplam Sembolü
$\mathbb{Q}$	Rasyonel Sayılar Kümesi
$\emptyset$	Boş Küme
$\forall$	Her
$\exists$	En az bir
$=$	Eşit
$\neq$	Eşit Değil
$\|\ \|$	Tam Değer
∞	Sonsuz

1. GİRİŞ

1.1. Tarihçe

Yazının icadından günümüze kadar, insanlar haberleşmelerinde gizliliği her zaman ön planda tutmuşlardır. İnsanlık tarihinden bu yana haberleşmek için yeni yöntemler geliştirilmiş ve iletişimleri diğer insanlardan gizleme gerekliliği doğduğundan itibaren de gizlilik, haberleşmedeki en önemli etken olmuştur. Bilgileri kodlamak, yüzlerce yıl önce ulusların ve imparatorlukların önemli ve gizli bilgileri yabancıların elinde geçmeden iletmek için ortaya çıkmıştır. Günümüzde ileti birçok farklı kaynaktan gönderilebilse de her zaman başkalarının eline geçme ihtimali vardır. Bu nedenden iletiyi kodlamak, yani kısaca bir kelimenin veya cümlenin başka bir kelime, sembol ya da sembol ile yerlerini değiştirerek gönderme, iletinin diğer insanlar tarafından anlaşılmasını zorlaştırmaktadır (Çimen vd 2007).

Herhangi bir ağın katılımcılar tarafından güvenli bir şekilde iletişim kurabilmesi için şifreleme algoritmaları kullanmaları gerekir. Şifreleme algoritmalarının birden fazla katılımcının olduğu ortamlarda kullanılmasındaki en önemli problem şifreleme anahtarının belirlenen ağın her katılımcısına doğru bir şekilde aktarılmasıdır. (Diffie and Hellman 1976).

Kriktografide bu süreç anahtar kurma protokolleri diye adlandırılır ve özellikle belirli aralıklarla şifreleme anahtarı güncelleyen sistemlerde hayati önem taşımaktadır. Anahtar kurma protokolleri iki gruba ayrılır, bu gruplar anahtar dağıtım ve anahtar anlaşma şeklindedir. Bunlar şifreleme anahtarlarının merkezi ya da merkezi olmayan yöntemler ile yapılmasına göre belirlenir. Yani, eğer belirlenen ağdaki katılımcıların hepsinin güvenilirliğine ikna olduğu üçüncü kişi (ya da kişiler) veya ağın içerisindeki bir katılımcı tarafından şifreleme anahtarı dağıtılıyorsa bu yöntem, anahtar dağıtma protokolleri şeklinde adlandırılır. Eğer, şifreleme anahtarı ağın içerisindeki katılımcıların kendileri tarafından oluşturdukları anahtarları birbirlerine yollayıp, yine katılımcıların önceden belirlediği bir fonksiyon ile hesaplanıyorsa buna anahtar anlaşma protokolleri denir. Anahtar anlaşması protokolü birinin bir başkasıyla güvenli bir şekilde veri alışverişi yapması gerektiğinde, önce paylaşılan bir anahtar oluşturması gerekir. Daha sonra mesaja erişmek isteyen bir düşman tarafından müdahale edilmesi

durumunda, güvenli olmayan bir kanalda bile güvenliğinden emin olmaları gerekir. Anahtar anlaşma protokolü, tek anahtarlı yapılarda iki tarafın gizli anahtar üzerinde anlaşması gereken durumlarda kullanılır. Ortam güvenli olmasa da bu protokoller önceden üzerinde anlaşılmış anahtara gerek duymaksızın tarafların güvenli bir şekilde gizli anahtar üzerinden anlaşmasını sağlar. Anahtar anlaşma protokolü, ilk olarak, 1976'da Whitfield Diffie ve Martin Hellman'le birlikte kaleme aldığı New Directions in Cryptography makalesi yeni bir anahtar değişim tekniğini gündeme getirmiş ve açık anahtarlı şifreleme algoritmalarının tasarlanmasını kolaylaştırmıştır. Diffie-Hellman gizli iletişimlerde kullanılabilecek ortak gizli anahtar üretir. Bu anahtar da ortak ağlarda (güvenli olmayan kanaldan) güvenli veri alışverişini sağlar.

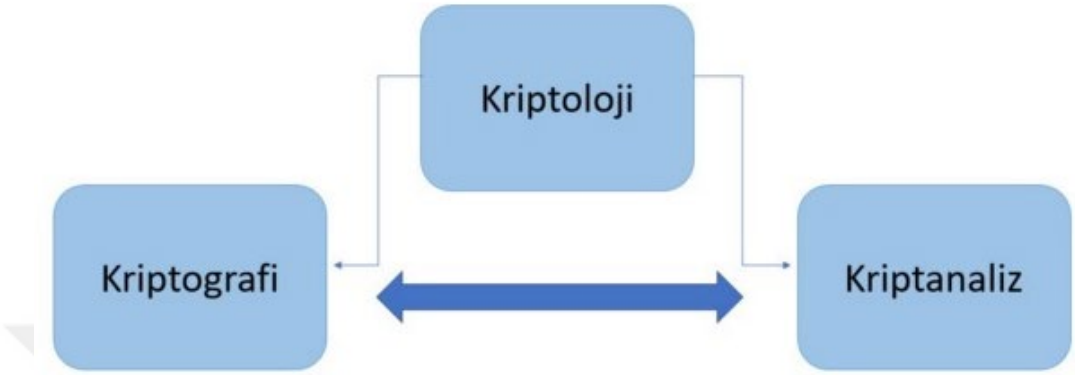
İki katılımcının herhangi bir merkezi otorite olmadan kendi anahtarlarını diğer katılımcıya açıklamadan ortak anahtarda karar kılınabileceği anlatılmıştır. Daha sonra, Ingemarsson, bu protokolü birden fazla katılımcının bulunduğu bir ağa uyarlayabilmek için ilk konferans anahtarı dağıtma protokolünü geliştirmiştir.

Kriptografi, köken olarak Yunanca gizli saklı anlamına gelen kryptos ve yazmak anlamına gelen graphein sözcüklerinden türetilmiştir. Kriptografi gizlilik, kimlik denetimi, bütünlük gibi bilgi güvenliği kavramlarını sağlamak için çalışan matematiksel yöntemler bütünüdür. Bu yöntemlerin amacı, iletinin gönderimi esnasında ortaya çıkabilecek saldırılardan iletiyi, ileti göndericisini ve alıcısını korumaktır. Yani kriptografi, okunabilir durumdaki bir iletinin istenmeyen kişiler tarafından okunamayacak bir hale dönüştürülmesinde kullanılan tekniklerin tümüdür (Çimen vd 2007).

Kriptanaliz; ele geçen şifrelenmiş, yani anlamsız bir metinden bazı teknikleri kullanarak doğru metni bulma yöntemi olarak adlandırılır (Çimen vd (2007). Analizcinin hedefi herhangi bir algoritma kullanılarak gizlenmiş metinleri açık metin haline dönüştürmektir. Bu hedefe algoritmada kullanılan gizli anahtarın tamamının veya belli bir kısmının elde edilebilmesiyle ulaşılır.

Kriptoloji; şifre bilimidir. Çeşitli iletilerin, yazıların belli bir sisteme göre şifrelenmesi, bu mesajların güvenlikli bir ortamda alıcıya iletilmesi ve iletilmiş mesajın deşifre edilmesidir.

Kriptanaliz ve **kriptografi, kriptoloji** adı altında toplanmaktadır. Bunlar arasındaki şema Şekil 1.1.1 de verilmiştir. (Demirel 2019).



Şekil 1.1.1. Kriptografi, Kriptoloji ve Kriptanaliz Arasındaki Şema

Kriptolojinin tarihine bakacak olursak, tarihinde iyi tanımlanmış üç aşama vardır.

Birinci aşama, konunun antik dönemdeki kökenlerinden başlayıp I. Dünya Savaşı boyunca devam eden manuel şifreleme dönemiydi. Bu aşama boyunca, kriptografi, bir kod yazmanının basit bir anımsatıcı yardımıyla makul bir şekilde yapabileceklerinin karmaşıklığıyla sınırlıydı. Sonuç olarak, şifreler boyut olarak en fazla birkaç sayfayla, yani yalnızca birkaç bin karakterle sınırlandırıldı. Hem kriptografi hem de kriptanaliz için genel ilkeler biliniyordu, ancak elde edilebilecek güvenlik her zaman manuel olarak yapılabileceklerle sınırlıydı. Bu nedenle, yeterli şifreli metin ve çaba verildiğinde çoğu sistem şifrelenebilir. Bu aşamayı düşünmenin bir yolu, iki bin yılda tasarlanan herhangi bir kriptografi şemasının kadim insanlar tarafından iyi kullanılmış olabileceğidir.

İkinci aşama, kriptografinin mekanizasyonu, Birinci Dünya Savaşı'ndan kısa bir süre sonra başladı ve bugün bile devam ediyor. Uygulanabilir teknoloji, telefon ve telgraf iletişimini veya Brunsvigas, Marchants, Facits ve Friedens gibi hesaplama makinelerini içeriyordu. Bu, II. Dünya Savaşı'nda tüm katılımcılar tarafından kullanılan rotor makinelerle sonuçlandı. Bu makineler, manuel olarak

mümkün olandan çok daha karmaşık işlemleri gerçekleştirebilir ve daha da önemlisi, daha hızlı, daha az hata şansı ile şifreleme ve şifre çözme yapabilirler. Şifrelerin güvenli boyutu buna göre büyüdü, böylece on binlerce, hatta yüz binlerce karakter uygulanabilir hale geldi. Elektromekanik cihazlardan elektronik cihazlara geçiş bu eğilimi hızlandırdı. Yalnızca seksen yılda kaydedilen ilerlemeyi göstermek için, 1999'da ABD hükümeti, saniyede 6,7 milyar bit kanıtlanmış bir iş hacmine sahip Veri Şifreleme Standardının (VŞS) tek bir silikon çip uygulamasını tasarladı ve üretti. Gelişmiş Şifreleme Standardı (GŞS) uygulanan 10 sapma tek bir silikon çip içinde İnternet omurga devresinde saniyede bit sayısı saniyede 10 gigabit olabilir. İlk mekanize şifreleme makineleri ile mümkün olan saniyede onlarca bit ile karşılaştırıldığında, birkaç saniyelik işlemde trilyonlarca şifre biti işlenebilir. 20. yüzyılın sonunda, tek bir iletişim kanalında ele alınması gereken şifreli metin hacmi neredeyse bir milyar kat arttı ve sürekli genişleyen bir hızla artmaya devam ediyor.

Üçüncü aşama; Yalnızca 20. yüzyılın son yirmi yılına dayanan, en radikal değişimi işaret ediyor. Kriptolojinin bilgi çağına dramatik uzantısı; dijital imzalar, kimlik doğrulama, kriptolojik işlevleri kullanmak için paylaşılan veya dağıtılmış yetenekler, vb. üzerindedir. Bu aşamayı açık anahtarlı kriptografinin görünümüyle eşitlemek cazip gelebilir, ancak bu çok dar bir görüş olur. Üçüncü aşama; tarihsel olarak somut belgelerin yardımıyla yapılan tüm işlevleri yerine getirmek için elektronik bilgilerin yollarını bulmanın kaçınılmaz sonucudur.

Düz bir metnin şifrelenmesi ve deşifrelenmesi kısaca aşağıda Şekil 1.1.2.'de gösterilebilir (Soyalıç 2004).

Şekil 1.1.2. Bir düz metnin şifrelenmesi ve deşifrelenmesi

1.2. Klasik Kriptografi

Klasik kriptografi matematiğe ve çok sayıda çarpanlara ayırmanın hesaplama zorluğuna dayanır. Klasik kriptografinin güvenliği, büyük sayının örneğin çarpanlara ayrılması için matematik probleminin yüksek karmaşıklığına dayanmaktadır.

Klasik kriptografide, orijinal veriler, yani düz metin, şifreli formata, yani şifreli metne dönüştürülür, böylece bu verileri güvenli olmayan iletişim kanallarından iletebiliriz. Verilerin düz metinden şifreli metne dönüşümünü kontrol etmek için anahtar olarak bilinen bir veri dizisi kullanılır. Bu düzenleme, şifreleme metninden orijinal bilgileri çıkarmak için anahtara ihtiyaç duyduğundan verileri güvende tutmaya yardımcı olur. Anahtar olmadan kimse verileri okuyamaz. Bu teknikte, tek yetkilinin alıcının anahtarına sahip olduğu varsayılır.

Klasik şifreleme teknikleri, Yer değiştirme (transposition) ve yerine koyma (substitution) yöntemleri olarak ikiye ayrılır (Kindap 2015).

a) Yer Değiştirme (Transposition) Yöntemi

Yer değiştirme tekniği, düz metin üzerinde permütasyonlar gerçekleştirerek, yani her tur için düz metnin her karakterini değiştirerek düz metni şifreli metne dönüştüren bir şifreleme tekniğidir. Sınırlı sayıda kelimeler içeren için bu yöntem az güvenlik sağlar. Örneğin, iki harften oluşan mesaj iki farklı, dört harften oluşan bir mesaj ise yirmi dört farklı şekilde sıralanabilir. Yani kelimede bulunan harflerin sayısının artması mesajın düşmanlar tarafından bulunmasını imkânsız kılar.

b) Yerine Koyma (Substitution) Yöntemi

Yerine Koyma yönetimi düz bir metindeki harflerin yerine başka harfler, sayılar ya da semboller koyarak yapılır. Yerine koyma şifreleri, aktarım şifreleri ile karşılaştırılabilir. Bir aktarım şifresinde, düz metnin birimleri farklı ve genellikle oldukça karmaşık bir sırayla yeniden düzenlenir, ancak birimlerin kendileri değişmeden bırakılır. Aksine, bir yerine koyma şifresinde, şifresiz metnin birimleri şifreli metinde aynı sırada tutulur, ancak birimlerin kendileri değiştirilir.

İki farklı yerine koyma şifresi vardır. Şifre tek harf üzerinde çalışıyorsa Tekli Alfabetik Yerine Koyma (Monoalphabetic) olarak, daha büyük harf grupları üzerinde çalışıyorsa Çoklu Alfabetik Yerine Koyma (Polyalphabetic) olarak adlandırılır.

- **Tekli Alfabetik Yerine Koyma (Monoalphabetic Substitution) Yöntemi**

Alfabede bulunan harflerin her seferde aynı harfle şifrelenmesidir. Bu yerine kullanma alfabesi olarak adlandırılır. Yerine kullanma alfabesi kaydırılabilir, tersine çevrilebilir veya daha karmaşık bir şekilde karıştırılabilir, bu durumda karışık alfabe veya dengesiz alfabe olarak adlandırılır. Geleneksel olarak, karışık alfabeler önce bir anahtar sözcük yazarak, içindeki tekrarlanan harfleri kaldırarak ve ardından alfabedeki kalan tüm harfleri olağan sırayla yazarak oluşturulabilir.

Bu sistemi kullandığınızda, " zebralar " anahtar kelimesi bize aşağıdaki alfabeleri verir:

Düz Metin: ABCDEFGHIJKLMNOPQRSTUVWXYZ
Şifreli Metin: ZEBRASCDFGHIJKLMNOPQTUVWXY

Genellikle şifreli metin noktalama işaretleri ve boşluklar çıkarılarak sabit uzunlukta bloklar halinde yazılır, bu kelime sınırlarını düz metinden gizlemek ve aktarım hatalarını önlemeye yardımcı olmak için yapılır. Bu bloklar "gruplar" olarak adlandırılır ve bazen bir "grup sayısı" ek bir kontrol olarak verilir. Mesajların telgrafla iletildiği zamandan kalma, genellikle beş harfli gruplar kullanılır:

SIAAZ QLKBA VAZOA RFPBL UAOAR

Mesajın uzunluğu beşe bölünemezse, sonunda " boşlar " ile doldurulabilir. Bunlar, apaçık saçmalıklara şifresini çözen herhangi bir karakter olabilir, böylece alıcı onları kolayca tespit edebilir ve atabilir.

- **Çoklu Alfabetik Yerine Koyma (Polyalphabetic Substitution) Yöntemi**

Polifabetik bir şifre, birden fazla yer değiştirme alfabesi kullanan, yer değiştirmeye dayalı herhangi bir şifredir. Çoklu alfabeli bir şifrede, birden çok şifreli alfabe kullanılır. Şifrelemeyi kolaylaştırmak için, tüm alfabeler genellikle geleneksel olarak tablo adı verilen büyük bir tabloya yazılır. Tablo genellikle 26 × 26'dır, bu

nedenle 26 tam şifreli metin alfabesi mevcuttur. Tabloyu doldurma ve daha sonra hangi alfabenin kullanılacağını seçme yöntemi, belirli çok alfabetik şifreyi tanımlar. Yeterince büyük düz metinler için yerine koyma alfabeleri tekrarlandığından, bu tür tüm şifrelerin kırılması bir zamanlar inanıldığından daha kolaydır.

En popüler olanlardan biri Blaise de Vigenère'dir. İlk olarak 1585'te yayınlandı, 1863'e kadar kırılmaz olarak kabul edildi ve aslında genellikle chiffre indéchiffrable ("çözülemez şifre" anlamına gelen Fransızca) olarak adlandırıldı.

Vigenere Şifreleme tablosunun ilk sıra düz metin alfabenin bir kopyasıyla doldurulursa ve ardışık satırlar soldaki bir yere kaydırılır. (Böyle basit bir tabloya tabula recta denir ve matematiksel olarak düz metin ve anahtar harflerin eklenmesine karşılık gelir) Daha sonra, hangi şifreli metin alfabesinin kullanılacağını seçmek için bir anahtar kelime kullanılır. Anahtar kelimenin her harfi sırayla kullanılır ve daha sonra baştan tekrarlanır. Dolayısıyla, anahtar kelime 'CAT' ise, düz metnin ilk harfi 'C', ikincisi 'A', üçüncüsü 'T', dördüncü 'C' alfabesi altında şifrelenir ve bu böyle devam eder. Pratikte, Vigenère anahtarları genellikle birkaç kelime uzunluğunda ifadelerdi. Vinegere Tablosu Şekil 1.2.1.'de aşağıda verilmiştir.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Şekil 1.2.1. Vinegere Tablosu

Şöyle bir örnek verilebilir:

Anahtar kelime : GEEKSFORGEEKS
Düz Metin : AYUSH
Şifreli Metin : GCYCZFMLYLEIM

- **Çoklu Harfle Şifreleme (Multiple – Letter Encryption) Yöntemi**

Çoklu şifreleme, önceden şifrelenmiş bir mesajı aynı veya farklı bir algoritma kullanarak bir veya daha fazla kez şifreleme işlemidir. Aynı zamanda kademeli şifreleme, kademeli şifreleme, çoklu şifreleme ve süper şifreleme olarak da bilinir. Süper şifreleme, çoklu şifrelemenin dış seviye şifrelemesini ifade eder.

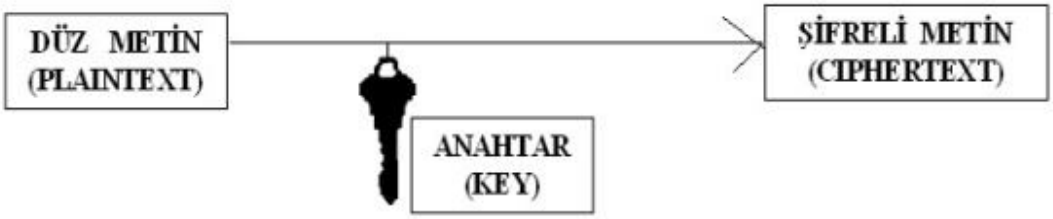
Johns Hopkins Üniversitesi'nden Matthew Green gibi bazı kriptograflar, çoklu şifrelemenin çoğunlukla var olmayan bir sorunu ele aldığını söylüyor: Modern şifreler nadiren kırılıyor. Kötü amaçlı yazılım veya bir uygulama hatası tarafından vurulma olasılığınız sizden çok daha yüksektir ve bu alıntıda çoklu şifrelemenin nedeni, yani zayıf uygulama yatıyor. İki farklı satıcıdan iki farklı şifreleme modülü ve anahtarlama işlemi kullanmak, güvenliğin başarısız olması için her iki satıcının ürününün de tehlikeye atılmasını gerektirir.

1.3. Modern Kriptografi

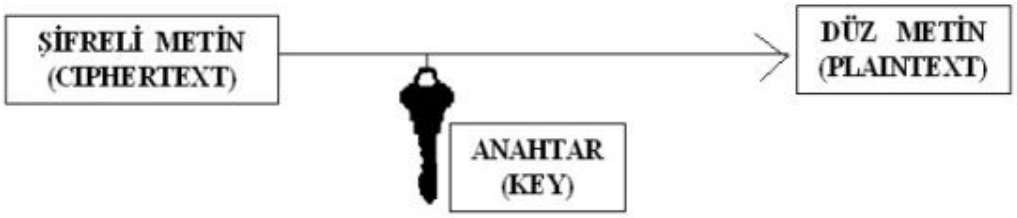
Eski Yunanlıların basit şifreleme yöntemlerinden günümüzün bilgisayarlı eliptik eğri algoritmalarına kadar hem bireyler hem de hükümetler tarafından güvenli mesajlar göndermek için çeşitli kodlar ve şifreler kullanılmıştır. Kişisel iletişimlerimizin ve verilerimizin artan miktarı çevrimiçine taşındıkça, internet güvenliğinin altında yatan fikirleri anlamak giderek daha önemli hale geldi. Modern Kriptografi tekniklerinden olan açık anahtarlı ve gizli anahtarlı şifreleme tekniklerini aşağıda açıklayalım.

a) Simetrik Anahtarlı Şifreleme (Symmetric Key Cipher)

Simetrik anahtarlı şifreleme sistemleri, bir mesajın şifrelenmesi ve şifresinin çözülmesi için aynı anahtarı kullanır, ancak bir mesaj veya mesaj grubu diğerlerinden farklı bir anahtara sahip olabilir. Simetrik şifrelerin önemli bir dezavantajı, onları güvenli bir şekilde kullanmak için gerekli olan anahtar yönetimidir. Her bir farklı iletişim tarafı çifti, ideal olarak, farklı bir anahtarı ve belki de değiştirilen her şifreli metin için paylaşmalıdır. Ağ üyelerinin sayısı arttıkça, gerekli anahtarların sayısı artar, bu da çok hızlı bir şekilde hepsini tutarlı ve gizli tutmak için karmaşık anahtar yönetimi şemaları gerektirir.



Şekil 1.3.1. Simetrik Anahtarlı Şifreleme (Kindap 2015)



Şekil 1.3.2. Simetrik Anahtarlı Şifre Çözme (Kindap 2015)

Simetrik anahtarlı şifreler Akış şifreler ve Blok şifreler olmak üzere ikiye ayrılır.

- **Akış Şifreler (Stream Ciphers)**

Akış şifreleri, düşük kaynak tüketimi ile hızlı uygulamalar için iyidir. Bu iki özellik, savunucunun blok şifreleme uygulaması için kaynaklara sahip olamayacak

cihazlarda direnç stratejileri uygulamasına yardımcı olur. Akış şifreleri, verileri daha büyük, sabit boyutlu parçalar halinde iletmekten daha doğal olarak bir akış modeline uyan kablosuz sinyalleri şifrelemek için de kullanışlıdır. Örneğin, A5 / 1 akış şifresi GSM telefonlarında kullanılır ve RC4 akış şifresi, kablosuz yerel alan ağları için güvenlik sisteminde kullanılmıştır.

One Time Pad ve RC4 yaygın olarak kullanılan bir akış şifreleridir. Akış şifreler bir blok şifresine göre daha hızlı sonuç verir. Akış şifreleme örneği aşağıda verilmiştir.

Şifreleme:

Düz metin $m = m_1 m_2 \ldots m_n$

Anahtar $k = k_1 k_2 \ldots k_n$

Şifreli Metin $c = c_1 c_2 \ldots c_n$

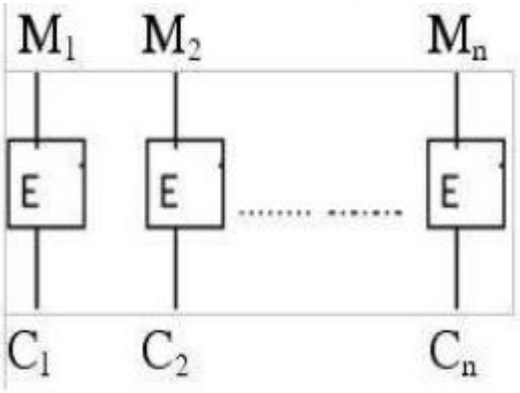
Burada her i için $c_i = m_i \oplus k_i$

- **Blok Şifreler (Block Ciphers)**

Bir blok şifreleme, akış şifrelerinde olduğu gibi her seferinde bir bit şifrelemek yerine, bir metin bloğunu şifrelemek için simetrik bir anahtarla birlikte deterministtik bir algoritma uygulayan bir şifreleme yöntemidir. Örneğin, ortak bir blok şifresi olan AES, 128 bitlik blokları önceden belirlenmiş uzunlukta bir anahtarla şifreler: 128, 192 veya 256 bit. Blok şifreleri, sabit boyutlu bit bloğu üzerinde çalışan sözde rasgele permütasyon (PRP) aileleridir. PRP'ler, tamamen rastgele permütasyonlardan ayırt edilemeyen ve bu nedenle güvenilir olmadığı kanıtlanana kadar güvenilir kabul edilen işlevlerdir.

Aynı metin bloklarının şifrelenmesi olasılığını ortadan kaldırmak için blok şifreleme modları geliştirilmiştir, önceki şifrelenmiş bloktan oluşturulan şifreli metin bir sonraki bloğa uygulanır. Bir başlatma vektörü olarak adlandırılan bir bit bloğu, aynı düz metin mesajı birkaç kez şifrelenmiş olsa bile şifreli metinlerin farklı kalmasını sağlamak için çalışma modları tarafından da kullanılır.

Blok şifreleri için çeşitli çalışma modlarından bazıları, diğerleri arasında CBC (şifre blok zinciri), CFB (şifre geri bildirimi), CTR (sayaç) ve GCM (Galois / Sayaç Modu) içerir. Blok şifreleme örneği Şekil 1.3.3.'de verilmiştir.



Şekil 1.3.3. Blok ve Şifreler (Kindap 2015)

b) Açık Anahtarlı Şifreleme (Public Key Cipher)

Açık anahtar şifrelemesi veya açık anahtar şifrelemesi, verileri iki farklı anahtarla şifrelemek ve anahtarlardan birini, açık anahtarı herkesin kullanımına açık hale getirmek için kullanılan bir yöntemdir. Diğer anahtar, özel anahtar olarak bilinir. Genel anahtarla şifrelenen verilerin şifresi yalnızca özel anahtarla çözülebilir ve özel anahtarla şifrelenen verilerin şifresi yalnızca genel anahtarla çözülebilir. Açık anahtar şifreleme, asimetrik şifreleme olarak da bilinir. Özellikle HTTPS'yi mümkün kılan TLS/SSL için yaygın olarak kullanılmaktadır.

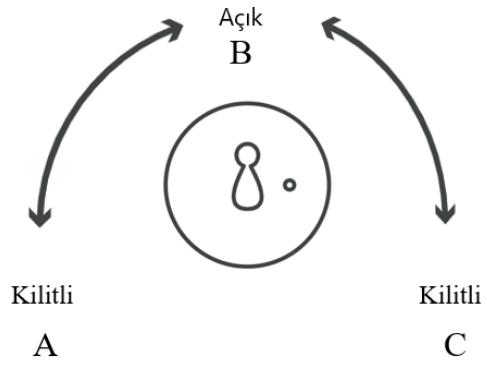
- **Açık Anahtarlı Sistemin Çalışma Mantığı**

Açık anahtarlı kriptografi, yeni başlayanlar için karmaşık görünebilir. Açık anahtarlı sistemin çalışma mantığı için Panayotis Vryonis adlı bir yazar kabaca aşağıdaki gibi bir benzetme yaptı. İki kişi olan Bob ve Alice'in belgeleri ileri geri göndermek için kullandıkları kilitli bir sandık hayal edelim. Normal bir kilidin yalnızca kilitli ve açık olmak üzere iki durumu vardır. Anahtarın bir kopyasına sahip olan herkes, sandık kilitliyse kilidini açabilir veya bunun tersi de yapabilir. Bob sandığı kilitleyip Alice'e gönderdiğinde, Alice'in sandığın kilidini açmak için anahtarın

kopyasını kullanabileceğini bilir. Esasen simetrik kriptografi olarak bilinen şey budur. Hem şifreleme hem de şifre çözme için bir gizli anahtar kullanılır ve bir tarafların her ikisi de aynı anahtarı kullanır.

Şimdi, bunun yerine Bob'un özel bir kilitle bir sandık yaptığını hayal edin. Bu kilidin iki yerine üç durumu vardır:

- A. Kilitli, anahtar tamamen sola döndü
- B. Ortada kilitsiz.
- C. Kilitli, anahtar tamamen sağa döndü.



Bir anahtar yerine iki anahtar bu kilitle birlikte gelir:

- Anahtar No. 1 yalnızca sola dönebilir
- Anahtar No. 2 yalnızca sağa dönebilir

Bu, sandık kilitliyse ve anahtar A konumuna çevrilirse, yalnızca 2 numaralı anahtar kilidi sağa, B konumuna döndürülerek kilidi açabilir. Sandık C konumunda kilitliyse, yalnızca 1 numaralı anahtar kilidi sola, B konumuna çevirerek açabilir.

Diğer bir deyişle, her iki anahtar da sandığı kilitleyebilir ancak bir kez kilitlendiğinde, yalnızca diğer anahtar kilidi açabilir.

Şimdi, Bob'un sadece sağa dönen anahtar olan 2 numaralı anahtarın birkaç düzine kopyasını çıkardığını ve bunları tanıdığı herkesle ve bir kopya isteyen herkesle paylaşarak bunu genel anahtarı haline getirdiğini varsayalım. 1 numaralı anahtarı kendisi için saklar bu onun özel anahtarıdır. Bu ne anlama gelir.

1. Alice, Bob'a gizli verileri sandık yoluyla gönderebilir ve sadece Bob'un kilidini açabileceğinden emin olabilir. Alice sandığı soldan sağa dönen genel anahtarla kilitlediğinde, yalnızca sağdan sola dönebilen bir anahtar kilidi açabilir. Bu, yalnızca Bob'un özel anahtarının kilidini açabileceği anlamına gelir.

2. Alice, eğer özel anahtarıyla kilitlenmişse, bagajın aslında Bob'dan geldiğinden ve bir taklitçi olmadığından emin olabilir. Bagajı kilitleyebilen ve böylece kilidin A konumunda olmasını veya tamamen sola dönmesini sağlayan tek bir anahtar vardır bu anahtar Bob'un özel anahtarıdır. Herkes anahtarı sağa çevirerek açık anahtarla kilidini açabilir, ancak sandığın Bob'dan olduğu garanti edebilir.

Bu benzetmeden ana hat için düz metin verilerini ve fiziksel anahtarlar için şifreleme anahtarlarını değiştirin ve açık anahtarlı kriptografi bu şekilde çalışır. Yalnızca özel anahtarın sahibi, genel anahtarın şifresini çözmesi için verileri şifreleyebilir; bu arada, herkes verileri genel anahtarla şifreleyebilir, ancak yalnızca özel anahtarın sahibi şifresini çözebilir.

Bu nedenle, herkes gizli anahtar sahibine güvenli bir şekilde veri gönderebilir. Ayrıca, özel anahtarın sahibinden aldıkları verilerin aslında o kaynaktan geldiğini ve bir taklitçiden olmadığını herkes doğrulayabilir.

Şekil 1.3.4. Açık Anahtarlı Şifreleme (Kindap 2015)

Şekil 1.3.5. Açık Anahtarlı Şifre Çözme (Kindap 2015)

En yaygın Açık Anahtarlı Sistemlerden olan RSA Açık anahtarlı şifreleme, aşağıda anlatılmıştır.

RSA Algoritması

RSA, güvenliği tam sayıları çarpanlarına ayırmanın algoritmik zorluğuna dayanan bir tür açık anahtarlı şifreleme yöntemidir. 1978'de Ron Rivest, Adi Shamir ve Leonard Adleman tarafından bulunmuştur. Bir RSA kullanıcısı iki büyük asal sayının çarpımını üretir ve seçtiği diğer bir değerle birlikte ortak anahtar olarak ilan eder. Seçilen asal çarpanları ise saklar. Ortak anahtarı kullanan biri herhangi bir mesajı şifreleyebilir, ancak şu anki yöntemlerle eğer ortak anahtar yeterince büyükse sadece asal çarpanları bilen kişi bu mesajı çözebilir. RSA şifrelemeyi kırmanın çarpanlara ayırma problemini kırmak kadar zor olup olmadığı hala kesinleşmemiş bir problemdir.

2. KURAMSAL TEMELLER

2.1. Grup Teorik Temeller

Tanım 2.1.1. $n \geq 2$ olmak üzere $A_1, A_2, \ldots A_n$ kümeleri verilsin. O zaman $A_1 \times A_2 \times \ldots . \times A_n$ Kartezyen çarpımının herhangi bir alt kümesi R'ye $A_1, A_2, \ldots A_n$ üzerinde bir n-li bağıntı denir.

Tanım 2.1.2. A ve B kümeleri verilsin ve f, A dan B ye bir bağıntı olsun. Eğer her $a \in A$ için $(a, b) \in f$ olacak biçimde bir tek $b \in B$ varsa f ye A dan B ye bir fonksiyon denir. Ayrıca A ya f nin tanım kümesi B yede f nin değer kümesi denir.

Tanım 2.1.3. A boş olmayan bir küme olsun. Herhangi bir $*: A \times A \to A$ fonksiyonuna A üzerinde bir ikili işlem denir ve her $a, b \in A$ için (a,b) elemanının $*$ altındaki görüntüsü $*(a, b) = a * b$ ile gösterilir.

Tanım 2.1.4. G boş olmayan bir küme ve G üzerinde bir $*$ ikili işlemi tanımlı olsun. Eğer $*$ işlemi aşağıdaki özellikleri sağlar ise $(G,*)$ sıralı ikilisine grup denir.

- $\forall\, a, b, c \in G$ için $(a * b) * c = a * (b * c)$ (Birleşme)
- $\forall\, a \in G$ için $(a * e) = (e * a)$ olacak şekilde bir $e \in G$ varsa, (e, G nin birim elemanı)
- $\forall\, a \in G$ için $(a * a') = (a' * a) = e$ olacak şekilde bir $a' \in G$ varsa, (a', a'nın ters elemanı)

Eğer $(G,*)$ grubunda fazladan $\forall\, a, b \in G$ için;

$$(a * b) = (b * a)$$

ise bu gruba değişmeli grup denir.

Tanım 2.1.5. G bir grup ve A, G nin boş olmayan alt kümesi olsun. Eğer H, G nin işlemine göre kapalı ve bu işleme göre bir grup ise o zaman H ya G nin bir alt grubu denir ve $H \leq G$ veya $G \geq H$ ile gösterilir.

Tanım 2.1.6. G bir olsun. Eğer $G = < a >$ olacak şekilde bir $a \in G$ varsa G ye a tarafından üretilen devirli grup denir.

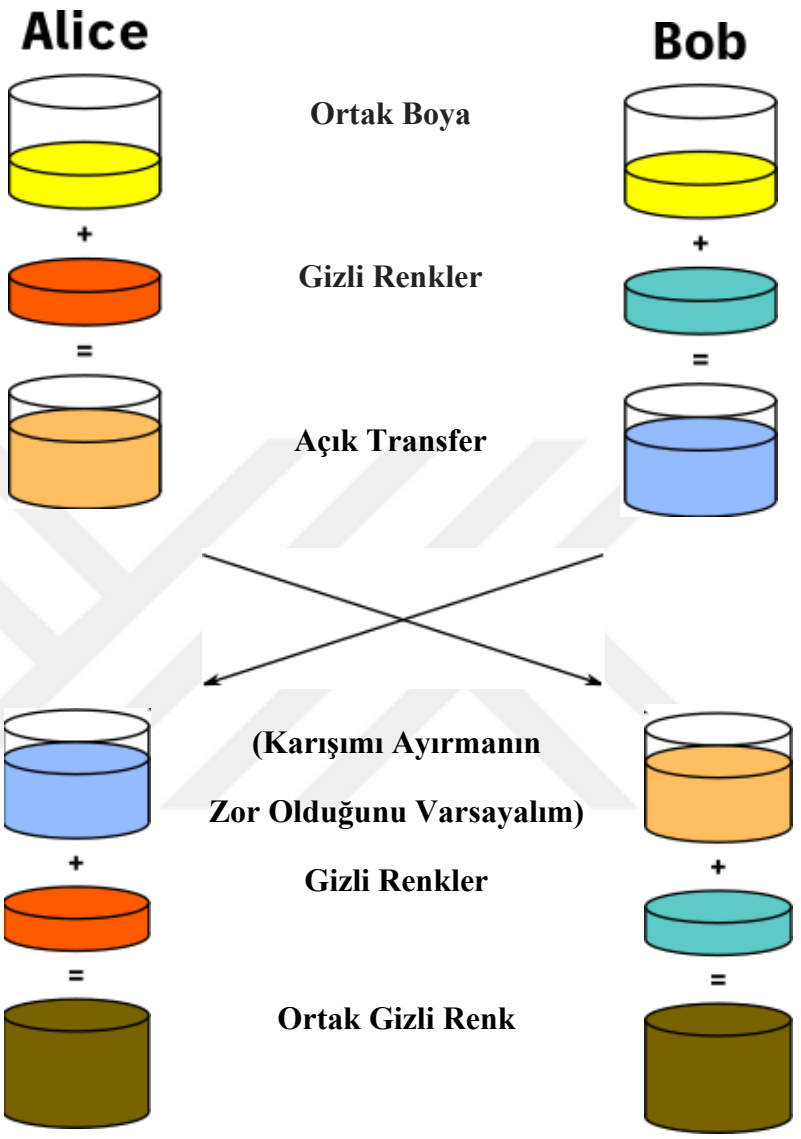
2.2. Anahtar Anlaşması

Anahtar anlaşması protokolleri, iki veya daha fazla tarafın, değiştirmeyi planladıkları verileri şifrelemek veya imzalamak için kullanabilecekleri paylaşılan bir şifreleme anahtarı oluşturmasına olanak tanır. Anahtar değişim protokolleri, bu amaca ulaşmak için genellikle farklı kriptografik teknikler kullanılabilir.

İki tarafın gizli bir şekilde iletişim kurması için, önce mesajları şifrelemek ve şifresini çözmek için kullanılacak gizli anahtarı değiştirmeleri gerekir. Bu ilk değişim şifreleme anahtarı, anahtar değişimi olarak adlandırılır.

Aşağıdaki diyagram anahtar değişiminin genel çalışma mantığını çok büyük sayılar yerine renkler kullanarak açıklar. Bu sürecin önemli bir parçası Alice ve Bob kendi gizli renklerini sadece karışım içinde değişirler. Sonunda her iki taraf matematiksel olarak arada dinleyen başka bir kişi tarafından geri döndürülmesi zor olan (bugünkü süper bilgisayarların mantıklı bir zamanda geri döndürememesi)aynı anahtarı elde eder. Bu aşamadan sonra Alice ve Bob oluşturmuş oldukları ortak gizli anahtarla aralarındaki veri alışverişini şifrelemek ve deşifrelemek için kullanırlar.

Sarı rengin zaten Alice ve Bob tarafından ortak olarak bilindiğini varsayalım



Şekil 2.2.1. Alice ve Bob Arasında Geçen Anahtar Anlaşması

a) Temel İfadeler

Temel anlaşma protokolleri konusundaki tartışmalarımız için gerekli olacak bazı temel kavramları tanımlayarak başlıyoruz.

Kriptografik kullanıma yönelik paylaşılan bir sırrın iki veya daha fazla tarafın kullanımına sunulduğu bir protokole anahtar oluşturma protokolü denir. Anahtar

oluşturma protokolleri, genellikle oturum anahtarları olarak adlandırılan paylaşılan sırlarla sonuçlanır. Bir oturum anahtarının genellikle geçici bir sır olması, yani yalnızca kısa bir süre veya oturum için kullanılacak ve ardından güvenli bir şekilde silinecek gizli bir değer olması amaçlanır.

Anahtar kuruluş protokolleri sınıfı, aşağıdaki gibi tanımlanan temel taşıma protokollerine ve anahtar anlaşma protokollerine bölünebilir.

Tanım 2.2.1. Anahtar taşıma protokolü, bir tarafın gizli bir değer yarattığı veya başka şekilde elde ettiği ve bunu diğerlerine güvenli bir şekilde aktardığı bir anahtar oluşturma protokolüdür (Caroline 2006).

Tanım 2.2.2. Anahtar anlaşma protokolü, paylaşılan bir sırrın iki (veya daha fazla) kişi tarafından bunların her birileri katkıda bulunan veya bunlarla ilişkili bilgilerin bir işlevi olarak türetildiği (ideal olarak) bir anahtar oluşturma tekniğidir. Taraf, ortaya çıkan değeri önceden belirleyebilir (Caroline 2006).

Bu tezde, anahtar taşıma protokolünden ziyade anahtar anlaşmayla ilgileniyoruz, bu nedenle bundan sonra anahtar anlaşmaya odaklanacağız.

Bir kanaldaki iletişimi tartışırken, aşağıdaki terminolojiye ihtiyacımız var:

Geçişler: Bir protokoldeki geçiş sayısı, protokolde alınıp verilen toplam mesaj sayısıdır.

Yayın: Yayın mesajı, bir protokoldeki her tarafa gönderilen bir mesajdır.

Tur: Bir tur, bir protokolde paralel olarak gönderilebilen ve alınabilen tüm mesajlardan oluşur.

Bu kavramlar, protokolün çalıştığı ağ türünden etkilenebilir.

Örneğin, tüm şebekeler mesajların yayınlanmasına izin vermez ve bu durumda tüm taraflara ayrı bir mesaj gönderilmelidir.

Bazen iki veya daha fazla katılımcı arasında bir protokolün genel olarak yürütülmesini bir protokol çalıştırması olarak adlandırırız.

Not 1: Kriptografik protokolün düşmanını veya saldırganını, protokolün amaçlanan güvenlik hedefini alt etmeye çalışan bir varlık olarak tanımlıyoruz. Pasif bir düşman, yalnızca iletişim kanallarını izleyen olandır. Aktif bir düşman, bir kanaldaki aktarımları silmeye, eklemeye veya bir şekilde değiştirmeye çalışan bir düşmandır. Aktif hasım tarafından yapılan bu tür saldırılara aktif saldırılar diyoruz.

Not 2: Genellikle protokol mesajlarının, aktif bir düşman tarafından saldırılara karşı korumasız kanallar veya ağlar üzerinden iletildiği varsayılır. Bu nedenle, bir protokolün güvenliğini analiz ederken, düşmanın, geçmiş mesajları kaydetme, değiştirme, silme, ekleme, yeniden yönlendirme, yeniden sıralama, yeniden dinleme ve yeni mesajlar enjekte etme yeteneği ile ağ üzerinde tam kontrole sahip olduğunu varsayıyoruz. Buna ek olarak, bir düşmanın yeni protokol uygulamaları başlatarak şüphelenmeyen yetkili taraflarla etkileşime geçebileceğini varsaymak yaygındır (Caroline 2006).

b) Diffie-Hellman Protokolü

Diffie-Hellman anahtar anlaşma protokolü, anahtar anlaşma protokolleri oluşturmak için temel bir teknik sunarak kriptografide yeni bir yol açtı. Diffie-Hellman anahtar anlaşması, anahtar dağıtım sorununa ilk pratik çözümü sağladı ve daha önce ortak bir gizli anahtarı olmayan iki tarafın, açık bir kanal üzerinden mesaj alışverişi yaparak böyle bir paylaşılan gizli anahtarı oluşturmasına izin verdi.

Protokol 1, iki varlık A ve B arasındaki orijinal Diffie-Hellman protokolünü tanımlar.

Protokol 1: Diffie-Hellman protokolü

A ve B, uygun (büyük) bir asal p seçer ve g, Z_p^* grubunun bir üreteci olsun.

Bir oturum anahtarı her gerektiğinde aşağıdaki adımlar uygulanmalıdır:

1. A rastgele bir a tamsayısı seçer, $1 \leq a \leq p-2$,

2. B rastgele bir b tamsayısı seçer, $1 \leq b \leq p-2$,

A ve B daha sonra aşağıdaki mesajları herhangi bir sırayla değiş tokuş yapar;

$$A \rightarrow B : \ g^a \ mod \ p$$

$$B \rightarrow A : \ g^b \ mod \ p$$

$g^b mod\ p$ mesajı aldığında A, $K_A = (g^b mod\ p)^{\ a} \ mod\ p$'yi, $g^a mod\ p$ mesajı alındığında B, $K_A = (g^a mod\ p)^{\ b} mod\ p$'yi hesaplar.

A ve B arasında paylaşılan gizli bir oturum anahtarı olarak kullanılabilen;

$K_A = K_B = K = g^{ab} \ mod \ p$ bulunur. Geçici değerler a ve b protokolünün tamamlanmasıyla silinir.

Sezgisel olarak, Protokol 1'in güvenliği hesaplamalı Diffie-Hellman (CDH) problemi ile ilişkili görünmektedir çünkü pasif bir düşman, oturum anahtarını belirlemek için CDH problemini çözmek zorunda kalacaktır. Aslında, düşmanın görevinin oturum anahtarlarını rastgele dizilerden ayırmak olduğu uygun bir güvenlik modelinde analiz edildiğinde, bu protokolün güvenliği, kararla ilgili Diffie-Hellman sorunuyla ilgilidir ve yalnızca pasif düşmanlara karşı ortaya çıkan anahtarın gizliliğini sağlar.

Orijinal Diffie-Hellman protokolü, gerçekte aktif düşmanlara karşı güvensizdir çünkü ne A ne de B, aldıkları mesajların kaynağı veya ortaya çıkan anahtarı paylaştıkları tarafın kimliği konusunda herhangi bir güvenceye sahip değildir. Bu, ortadaki adam saldırısı olarak bilinen orijinal Diffie-Hellman protokolüne iyi bilinen bir saldırı ile kanıtlanmıştır.

Ortadaki Adam Saldırıları

Protokol 1'e yapılan ortadaki adam saldırısı şu şekilde işliyor. Protokol 1'e gelince A ve B'nin uygun (büyük) bir asal p ve g, Z_p^* grubunun üreteci olsun, aşağıdaki adımlar uygulanmalıdır:

1. A rastgele bir a tamsayısı seçer, $1 \leq a \leq p-2$,

2. B rastgele bir b tamsayısı seçer, $1 \leq b \leq p-2$,

Rakip olan E, bu protokol çalıştırmasında ortadaki adam saldırısı başlatmak isterse;

E geçici olarak rasgele a' ve b' $1 \leq a', b' \leq p-2$, sayılarını seçer, daha sonra E A'nın B'ye gönderdiği mesajları keser ve değiştirerek B'ye iletir, ayni şekilde bu işlemi B'nin A'ya gönderdiği mesaj içinde yapar.

A		E		B
$g^a \ mod \ p$	$\rightarrow$	$g^{a'} \ mod \ p$	$\rightarrow$	
	$\leftarrow$	$g^{b'} \ mod \ p$	$\leftarrow$	$g^b \ mod \ p$

$g^{b'} \ mod \ p$ mesajı alındığında, A $K_A = (\ g^{b'} \ mod \ p)^a \ mod \ p$ mesajını ve E $K_{EA} = (\ g^a \ mod \ p)^{b'} \ mod \ p$ mesajını hesaplar.

$g^{a'} \ mod \ p$ mesajı alındığında, B $K_B = (\ g^{a'} \ mod \ p)^b \ mod \ p$ mesajını ve E $K_{EB} = (\ g^b \ mod \ p)^{a'} \ mod \ p$ mesajını hesaplar.

A ve B anahtarı paylaştıklarına inanmalarına rağmen $K_A \neq K_B$ olamadığından anahtarı paylaşamamışlardır. Bunun yerine A, E ile $K_A = K_{EA}$ ve B, E ile $K_B = K_{EB}$ mesajlarını paylaşmışlardır.

Artık A, B'ye K_A ile şifrelenmiş bir mesaj gönderdiğinde, E bunu K_{EA} ile çözebilir K_{EB} ile yeniden şifreleyebilir ve B'ye gönderebilir. Benzer şekilde, E, B'den A'ya K_{EB} ile gönderilen mesajların şifresini çözebilir, onları yeniden şifreleyebilir. K_{EA} ile görüşün ve onları A'ya gönderin. Bu şekilde A ve B güvenli bir kanalı paylaştıklarına inanırken, aslında E aralarındaki tüm iletişimi kontrol ediyor (Caroline 2006).

c) Kimliği Doğrulanmış Anahtar Anlaşması

Bu kısım Caroline (2006) dan esinlenilerek hazırlanmıştır.

Protokol 1, kimliği doğrulanmadığı için ortadaki adam saldırısına karşı savunmasızdır. Bununla, protokole katılanların, ortaya çıkan anahtarı paylaşabilecekleri başkalarının kimliklerini doğrulamanın hiçbir yolu olmadığını söylüyoruz.

Artık, önemli anlaşma protokolleri için bazı kimlik doğrulama kavramlarını gayri resmi olarak tanımlayacağız. Aşağıdaki kavramlar genel olarak temel kuruluş

protokolleri için geçerli olsa da, biz sadece ana anlaşma protokolleriyle ilgiliyiz ve bu nedenle aşağıdaki tanımlarımızı bu durumla sınırlayacağız.

Tanım 2.2.3. Anahtar kimlik doğrulaması, bir tarafın, özel olarak tanımlanmış bir ikinci taraf (ve muhtemelen ek olarak tanımlanan güvenilir taraflar) dışındaki hiçbir tarafın belirli bir gizli anahtara erişemeyeceğinin garanti edildiği özelliktir (Caroline 2006).

Tanım 2.2.4. Kimliği doğrulanmış bir anahtar anlaşması protokolü, anahtar kimlik doğrulaması sağlayan bir anahtar anlaşma protokolüdür (Caroline 2006).

A işletmesinin kimliği doğrulanmış bir anahtar anlaşma protokolünü çalıştırdığını varsayıyoruz. Anahtar kimlik doğrulaması özelliği, ikinci tarafın (örneğin B) gerçekten gizli anahtara sahip olduğunu veya B'nin protokol çalıştırmasına dahil olduğunu bile garanti etmez. Bununla birlikte, A dışındaki herhangi bir varlık gizli anahtarı hesaplayabilirse, o varlığın B olması gerektiğini garanti eder. Bu nedenle, anahtar kimlik doğrulaması bazen daha kesin olarak (örtük) anahtar kimlik doğrulaması olarak anılır (Caroline 2006).

Tanım 2.2.5. Anahtar onayı, bir tarafın ikinci (muhtemelen tanımlanamayan) bir tarafın gerçekten belirli bir gizli anahtara sahip olduğundan emin olduğu mülktür (Caroline 2006).

Pratikte, bir anahtara sahip olduğunuzu göstermenin, anahtarın kendisinin tek yönlü bir özetini üretmek veya anahtarı kullanarak bilinen bir değeri şifrelemek dahil olmak üzere çeşitli yolları vardır. Bu tür yöntemlerin dezavantajı, bilgi hesaplama açısından sınırlı bir düşman için yararlı olmasa bile, anahtarın değeri hakkında bazı bilgilerin açığa çıkmasıdır. Alternatif olarak, sıfır bilgi teknikleri bir anahtara sahip olduğunu gösterirken, değeri hakkında ek bilgi sağlamamak için kullanılabilir.

Tanım 2.2.6. Açık anahtar kimlik doğrulaması, hem (örtük) anahtar kimlik doğrulaması hem de anahtar onayı bekletildiğinde elde edilen özelliktir (Caroline 2006).

Tanım 2.2.7. Varlık kimlik doğrulaması, bir tarafın, bir protokole dahil olan ikinci bir tarafın kimliğinin ve ikinci tarafın protokole fiilen katıldığının güvence altına alındığı (doğrulayıcı kanıtların elde edilmesi yoluyla) süreçtir (Caroline 2006).

İkinci tarafın protokole fiilen katıldığına dair garanti, birinci tarafın aldığı doğrulayıcı kanıtın taze olmasını sağlar, yani bunun yeni bir kanıt olduğu ve ikinci tarafla daha önceki bazı etkileşimden (bazı yetkisiz kuruluşlar tarafından) tekrar oynatılmadığı anlamına gelir.

Varlık kimlik doğrulaması tüm protokollerde bir gereklilik değildir, ancak kimliği doğrulanmış anahtar anlaşma protokolleri oluşturmak için bir araç olarak kullanılabilir. Ancak bu durumda, böyle bir protokolde, kimliği doğrulanan tarafın, anahtar üzerinde mutabık kalınan tarafın aynı olması kritiktir.

Önemli bir anlaşmanın doğrulanmasının birçok yolu vardır. Örneğin, uzun vadeli bir sırrı paylaşan varlıklar, geçici bir gizli oturum anahtarı oluşturmak isteyebilir. Bu durumda varlıklar, bir anahtar anlaşma protokolünü doğrulamak için uzun vadeli paylaşılan sırrı kullanabilir. Alternatif bir yaklaşım, anahtar anlaşması protokolünü doğrulamak için açık anahtar şifreleme (her bir varlığın uzun vadeli bir genel ve özel anahtar çiftine sahip olduğu) kullanmaktır. Bu durumda, genel anahtarların kimliğini doğrulamak için sertifikalar gerekir.

- **Kimliği Doğrulanmış Diffie-Hellman Protokolleri**

Açık anahtar teknikleri kullanarak orijinal Diffie-Hellman protokolü (Protokol 1) için örtük kimlik doğrulama sağlayan bazı basit protokol örnekleri vereceğiz. Blake-Wilson ve diğerleri tarafından sunulan iki doğrulanmış anahtar anlaşma protokolünü açıklayacağız.

Protokol 3 'ü tanımlayarak başlıyoruz. Protokol 1'de olduğu gibi, açık bir kanal üzerinden iletişim kurabilen ve paylaşılan bir gizli oturum anahtarı oluşturmak isteyen A ve B varlıkları arasındaki protokolü tanımlarız. Protokol, kimlik doğrulama için genel anahtar tekniklerini kullandığından A ve B, sırasıyla $\langle X_A, x_A \rangle$ ve $\langle X_B, x_b \rangle$ açık ve gizli anahtar çiftlerini gerektirir.

P ve q büyük asal sayılar olduğunu varsayalım, burada $q|(q-1)$ ve g, q'uncu mertebeden Z_p^*'nin bir elemanıdır.

Ayrıca sabit bir l değeri (genellikle güvenlik parametresi) için $\boldsymbol{H}: \{\mathbf{0},\mathbf{1}\}^* \rightarrow \{\mathbf{0},\mathbf{1}\}^l$ fonksiyonunun kriptografik bir değeri olmak üzere x_A ve x_B, Z_q^* fonksiyonundan rastgele seçildiğinde $X_A = g^{xA}$ ve $X_B = g^{xB}$ olur.

Açıkladığımız bir sonraki Protokol 2 oturum anahtarının biraz farklı bir şekilde oluşturulması dışında yukarıdaki Protokol 1'e çok benzer. Değişiklik küçük gibi görünse de ortaya çıkan protokol Protokol 3, Protokol 2'den farklı güvenlik özelliklerine sahiptir.

Protokol 2:

Bir oturum anahtarı her gerektiğinde aşağıdaki adımlar uygulanmalıdır:

1. A rastgele geçici bir $a \in Z_q$ tamsayısı seçer.

2. B rastgele geçici bir $b \in Z_q$ tamsayısı seçer.

A ve B daha sonra aşağıdaki mesajları herhangi bir sırayla değiştirir:

$A \rightarrow B : T_A = g^a mod\ p$

$B \rightarrow A : T_B = g^b mod\ p$

$g^b mod\ p$ mesajı aldığında A, $K_A = H(T_B^a\ mod\ p, X_B^{XA} mod\ p)'yı$, $g^a mod\ p$ mesajı alındığında B, $K_B = H\left(T_A^b\ mod\ p, X_A^{XB} mod\ p\right)$'yi hesaplar.

A ve B arasında paylaşılan gizli bir oturum anahtarı olarak kullanılabilen

$K_A = K_B = K = H(g^{ab}\ mod\ p, g^{XAXB} mod\ p)$ bulunur. Geçici değerler a ve b protokolünün tamamlanmasıyla silinir.

Protokol 3:

Bir oturum anahtarı her gerektiğinde aşağıdaki adımlar uygulanmalıdır:

1. A rastgele geçici bir $a \in Z_q$ tamsayısı seçer.

2. B rastgele geçici bir $b \in Z_q$ tamsayısı seçer.

A ve B daha sonra aşağıdaki mesajları herhangi bir sırayla değiştirir:

$A \rightarrow B : T_A = g^a mod\ p$

$B \rightarrow A : T_B = g^b mod\ p$

$g^b mod\ p$ mesajı aldığında A, $K_A = H(T_B^{XA}\ mod\ p, X_B^a mod\ p)'yı$, $g^a mod\ p$ mesajı alındığında B, $K_B = H\left(X_A^b\ mod\ p, T_A^{XB} mod\ p\right)$'yi hesaplar.

A ve B arasında paylaşılan gizli bir oturum anahtarı olarak kullanılabilen

$K_A = K_B = K = H(g^{XAb}\ mod\ p, g^{XBa} mod\ p)$ bulunur. Geçici değerler a ve b protokolünün tamamlanmasıyla silinir.

Bir kez daha p ve q'nun büyük asal sayılar olduğunu varsayıyoruz; burada $q|(q-1)$ ve g, q'uncu mertebeden Z_p^*'nin bir elemanıdır. Ayrıca A ve B nin Protokol 2'de olduğu gibi üretilen sırasıyla $\langle X_A, x_A \rangle$ ve $\langle X_B, x_b \rangle$ açık ve gizli anahtar çiftlerine sahip olduğunu varsayıyoruz.

Protokol 3'te, mesaj akışlarının sırasının anahtarın hesaplanmasında önemli olduğu tespit edilmiştir. Protokolün başlatıcısı (bu durumda başlatıcı, protokoldeki ilk mesajı gönderdiği için A dır) oturum anahtarını yanıtlayandan farklı bir şekilde hesaplar (bu durumda B). Protokolü kimin başlattığı konusunda bir kafa karışıklığı varsa (Örneğin A ve B'nin ikisi de protokolü başlattıklarına inanırlar), o zaman A ve B aynı oturum anahtarını oluşturmayacaktır (Caroline 2006).

- **Protokol 2 ve 3'ün Güvenlik Özellikleri**

Şimdi, Protokol 2 ve 3'ün bilinen anahtar güvenliği, (mükemmel) iletme gizliliği ve anahtar güvenliği ihlal kimliğine bürünme saldırılarına karşı direnç gibi güvenlik özelliklerine sahip olup olmadığını gayri resmi olarak inceliyoruz. Protokol 2 ve 3 arasındaki açık benzerliğe rağmen, bu analizin sonuçları iki protokol arasındaki bazı farklılıkları açıkça göstermektedir.

Her bir protokolün neden her bir güvenlik özniteliğine sahip gibi görünüp görünmediğine dair resmi olmayan argümanlar vereceğiz.

Bilinen oturum anahtarı güvenliği: Protokol 2 ve 3'ün her ikisi de bilinen anahtar güvenliğine sahip gibi görünmektedir, çünkü bir düşmanın, önceki oturum anahtarları bilgisi göz önüne alındığında yeni bir oturum anahtarı hakkında bilgi edinmesi mümkün görünmemektedir. Bunun ana nedeni, oturum anahtarlarının bir

kriptografik karma işlevin çıktıları olarak üretilmesi ve bu karma işlevin girdilerinin kurulan her yeni oturum anahtarı için değişmesidir. Özet fonksiyonunun tek yönlü olduğunu varsayarsak, o zaman rakip özet fonksiyonunun girdilerini çıktıdan belirleyemez. Bu nedenle rakip, yeni bir oturum anahtarının değerini belirlemede yararlı olabilecek önceki oturum anahtarlarından herhangi bir bilgi öğrenmez.

(Mükemmel) ileri gizlilik: Protokol 2 mükemmel bir ileri gizliliğe sahip görünüyor. Bunun nedeni, bir rakip x_A ve x_B özel anahtarlarını bilse bile, eğer hesaplamalı Diffie-Hellman sorununun zor olduğunu varsayarsak, düşman M'yi T_A ve T_B değerlerinden hesaplayamaz.

Öte yandan, x_A ve x_B özel anahtarları ve T_A ve T_B değerleri verildiği için, rakip oturum anahtarını hesaplayabildiğinden, Protokol 3 mükemmel iletme gizliliğine sahip değildir. Bununla birlikte, bir düşman, oturum anahtarını hesaplamak için her iki özel anahtara da ihtiyaç duyacağından, Protokol 3, kısmi ileri gizliliğe sahip gibi görünmektedir. Yalnızca bir özel anahtarla, rakip, Özet fonksiyonu girdilerinden ikisini birden hesaplayamaz, yalnızca birini hesaplayabilir.

Anahtar anlaşmasında kimlik gizlemenin olmaması: Protokol 2, anahtar güvenliği ihlal kimliğine bürünme saldırılarına karşı dirençli değildir. A'nın özel anahtarı x_A verildiğinde, rakip E, bir $b \in Z_q$ değeri seçerek, $T_B = g^b mod\ p$ 'yi hesaplayarak ve bunu A'ya göndererek B gibi davranabilir. E, yanıt olarak A'nın T_A değerini alacaktır. Şimdi E, K_A değerini şu şekilde hesaplayabilir:

$H\left(X_B^{x_A}\ mod\ p, T_A^b mod\ p\right)$ ve E artık A ile bir anahtarı paylaşırken, A anahtarının B ile paylaşıldığına inanıyor. Bir rakibin x_A, x_B, T_A ve T_A değerlerinden bir oturum anahtarı hesaplaması mümkün değildir.

Yukarıda analiz edilen üç güvenlik özelliğinden, Protokol 2 ve 3'ün benzerliklerine rağmen gerçekten farklı güvenlik garantileri sağladığı görülebilir. Bu özelliklerin kurulması daha karmaşık analizler gerektirdiğinden, Protokol 2 ve 3'ün bilinmeyen anahtar paylaşım saldırılarına ve anahtar kontrolüne karşı dirençli olup olmadığını dikkate almıyoruz. Protokol 2 ve 3'ün güvenli kabul edilip edilmeyeceği sorusu kişinin güvenlik tanımına bağlıdır. Kimliği doğrulanmış anahtar anlaşma

protokolleri için iyi bir güvenlik tanımı oluşturmak önemsiz bir iş değildir. (Caroline 2006).

3. MATERYAL VE YÖNTEM

3.1. Grup Etkisi

Tanım 3.1.1. $(G,\cdot)$ bir grup ve Ω bir küme olsun. $G \times \Omega \to \Omega$ fonskiyonu bir dönüşüm olsun. ($\forall\, g \in G$ için ve $\omega \in \Omega$ için $g \otimes \omega$ yazılabilir) Bu dönüşüm aşağıdaki koşulları karşılıyorsa, G'nin Ω kümesi üzerine grup etkisi vardır denir.

(i) $\forall\, g_1, g_2 \in G\ ve\ \omega \in \Omega\ için\ g_1 \otimes (g_2 \otimes \omega) = (g_1 \cdot g_2) \otimes \omega\ (uyumluluk)$

(ii) $\forall\, \omega \in \Omega\ için\ e \otimes \omega = \omega,\ burada\ e \in G$ özdeşliktir.

Aşağıdaki örnekten önce başka bir tanım verelim.

$\mathbb{P}^1_{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$ kümesine projektif komplex doğru denir.

Örnek 3.1.2 $SO_2(R) \times \mathbb{P}^1_{\mathbb{C}} \to \mathbb{P}^1_{\mathbb{C}}$ ye olmak üzere;

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes z = \frac{az+b}{cz+d}$$

şeklinde tanımlansın. $\mathbb{P}^1_{\mathbb{C}}$ yukarıdaki özelliği sağlıyorsa $SO_2(R)$ üzerinde bir etkidir.

(i) koşulunun karşılandığını göstereceğiz.

$\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \begin{pmatrix} e & f \\ g & h \end{pmatrix} \in SO_2(R)\ ve\ z \in \mathbb{P}^1_{\mathbb{C}}$ olsun;

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes \left(\begin{pmatrix} e & f \\ g & h \end{pmatrix} \otimes z \right) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \otimes \frac{ez+f}{gz+h}$$

$$= \frac{a\frac{ez+f}{gz+h} + b}{c\frac{ez+f}{gz+h} + d}$$

$$= \frac{a(ez+f) + b(gz+h)}{c(ez+f) + d(gz+h)}$$

$$= \frac{(ae+bg)z+(af+bh)}{(ce+dg)z+(cf+dh)}$$

$$= \begin{pmatrix} ae+bg & af+bh \\ ce+dg & cf+dh \end{pmatrix} \otimes z$$

$$= \left(\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} e & f \\ g & h \end{pmatrix} \right) \otimes z.$$

$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes z = z$ olduğundan (ii) koşulu sağlanır (Çağman vd 2020).

3.2. Ortogonal Gruplar

a) O_n

Tanım 3.2.1. Özel olarak $\mathcal{H} = R^n$ alalım. $x = (x_1, \ldots . x_n)$, $y = (y_1, \ldots . y_n)$

$\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n$

$U(\mathcal{H}) = U(R^n) = \{T: R^n \to R^n$ sürekli lineer ve tersinir dönüşüm ve $\langle T(x), T(y) \rangle = \langle x, y \rangle\}$ $T \in U(R^n)$ ise

$x, y \in R^n \; \|T(x) - T(y)\| = \sqrt{T(x) - T(y)} = \sqrt{\langle x, y \rangle} = \|x - y\|$

T bir izometridir. Ayrıca T lineer olduğundan $T(0) = 0$ dolayısıyla

$x = (x_1, \ldots . x_n) \in R^n \; T(x) = Ax$ olacak şekilde $A = A_{n \times n}$ olacak şekilde ortoganal matrisi vardır. $A_{n \times n}$: ortogonal matris.

Bu durumda $U(\mathcal{H}) = O(n, r) = O(n) = \{A_{n \times n}: A$ ortogonal matris $(A^t A = A A^t = I)\}$ grubuna ortogonal grup denir.

$O(n) = \{A_{n \times n}: A$ ortogonal matris$\}$ (A ortogonal $\Rightarrow \det A = \mp 1$)

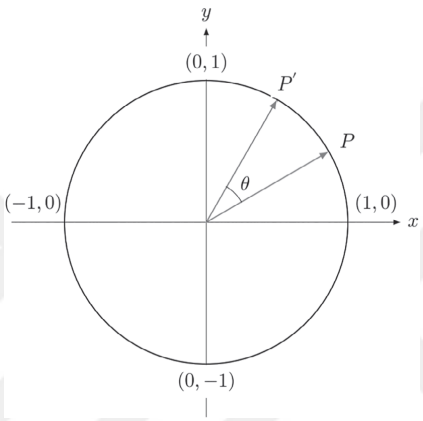
b) SO_n

Özel ortogonal grup $SO_n(R)$, her bir elemanı determinantı 1 olan $O_n(R)$ nin bir alt grubudur,

yani;

$$SO_n(R) = \{M \in O_n(R) | \det(M) = 1\}$$

olur.



Şekil 3.2.1. Bir P vektörünün θ açısıyla dönüşü (Çağman vd 2020)

Özellikle, $SO_2(R)$ gerçek düzlemdeki tüm uygun dönüşlerin grubudur. R_θ Şekil 1'deki gibi θ açısıyla dönme olsun. Ardından R_θ, (1, 0) noktasını (cosθ, sinθ) ve (0, 1) ile (−sinθ, cosθ) ile eşler. Yani, gerçek düzlem için birimdik taban $\beta = \{(1,0),(0,1)\}$'ya göre, R_θ'nın matris temsilinin $M = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ olduğunu görebiliriz. Böylece $M \in O_2(R)$ olur.

Çünkü;

$$MM^T = \begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix}\begin{pmatrix} cos\,\theta & sin\,\theta \\ -sin\,\theta & cos\,\theta \end{pmatrix}$$

$$= \begin{pmatrix} cos^2\theta + sin^2\theta & cos\,\theta sin\,\theta - cos\,\theta sin\,\theta \\ sin\,\theta cos\,\theta - cos\,\theta sin\,\theta & sin^2\theta + cos^2\theta \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

dir. Ek olarak, detM $= 1$ olduğuna dikkat etmek gerekir. Dolayısıyla, $M \in SO_2(R)$ gerçel düzlemde keyfi bir dönüşün $SO_2(R)$ 'nin bir elemanı olduğu anlamına gelir.

Şimdi, $SO_2(R)$ 'nin keyfi bir M elemanını ele alıyoruz. M sütunlarının gerçek düzlem için ortonormal bir taban oluşturduğunu görmek kolaydır. Yani gerçel düzlemde, M'nin sütunları orijini merkezli birim çember üzerinde ortogonal vektörler olarak bulunur. $U = (cos\,\theta, sin\,\theta)$ vektörünü orijin merkezli birim çemberin parametrik temsili olarak düşünürsek, sadece u vektörüne ortogonal olan orijin merkezli birim çember üzerinde $v = (-sin\,\theta, cos\,\theta)$ ve $w = (sin\,\theta, -cos\,\theta)$ vektörlerini elde ederiz.

Sütunlar olarak u ve v vektörleri kullanılarak oluşturulan $\begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix}$ matrisi, $SO_2(R)$ 'nin bir elemanıdır çünkü daha önce tanımlanan $M \in SO_2(R)$ matrisi ile aynıdır, fakat u ve w vektörlerini sütun olarak döndürerek oluşturulan $\begin{pmatrix} cos\,\theta & sin\,\theta \\ sin\,\theta & -cos\,\theta \end{pmatrix}$ matrisi $SO_2(R)$'nin bir elemanı değildir, çünkü bu matrisin determinantı 1'e eşit değildir.

Bu nedenle, $M \in SO_2(R)$ 'nin tüm elemanları Tanım 2.1. formunda olmalıdır. $M \in SO_2(R)$ değişmeli bir gruptur, yani $SO_2(R)$ elemanlarından rastgele seçilen bir M, N çifti için MN = NM olur.

Aslında, M ve N matrislerini sırasıyla, $\begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix}$ ve $\begin{pmatrix} cos\,\phi & -sin\,\phi \\ sin\,\phi & cos\,\phi \end{pmatrix}$olarak alırsak;

$$MN = \begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix}\begin{pmatrix} cos\,\phi & -sin\,\phi \\ sin\,\phi & cos\,\phi \end{pmatrix}$$

$$= \begin{pmatrix} cos\,\theta cos\,\phi - sin\,\theta sin\,\phi & -cos\,\theta sin\,\phi - \sin\theta cos\,\phi \\ \sin\theta cos\,\phi + cos\,\theta sin\,\phi & -sin\,\theta sin\,\phi + cos\,\theta sin\,\phi \end{pmatrix}$$

$$= \begin{pmatrix} cos\,\phi & -sin\,\phi \\ sin\,\phi & cos\,\phi \end{pmatrix}\begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix}$$

$$= NM$$

olur.

4. ARAŞTIRMA BULGULARI

4.1. Yarı Grup Etkisi Üzerine Yapılan Anahtar Anlaşması Protokolleri

Bu bölümümüzde genel yarı grup etkisine dayanan grup anahtarı yönetimi için bir protokol paketi sunuyoruz. Anahtarın inşası dağıtık ve işbirliğine dayalı bir şekilde yapılır. Bazı durumlarda önceki mevcut protokollerin güvenlik seviyesini ve iletişim ek yüklerini geliştirebilecek örnekler sağlanmıştır. Pasif saldırılara karşı güvenlik dikkate alınır ve herhangi bir özel senaryoda yarı grup etki probleminin zorluğuna bağlıdır. (Schnyder et all 2016).

A) Giriş

Anahtar değişimi için geleneksel kriptografik araçlar, iletişim süreci bir grup düğüm veya kullanıcı içinde yürütüldüğünde yararlı olmayabilir. Grup anahtar yönetimi için üç ana sınıfa bölünebilen birkaç yaklaşım vardır:

- Tek bir varlığın tüm grubu kontrol etmekten, depolama gereksinimlerini en aza indirmekten hem istemci hem de sunucu tarafındaki hesaplama gücünü ve iletişim genel giderlerinden sorumlu olduğu merkezi protokoller,
- İş yükünün tek bir noktada yoğunlaşmasını önlemek için büyük bir grubun alt gruplara ayrıldığı merkezi olmayan,
- Anahtar üretiminin dağıtık ve iş birliğine dayalı bir şekilde gerçekleştirildiği yer (Juan Antonio et all 2015).

Muhtemelen hafif ve mobil cihazlardan oluşan bir dizi düğümün bir ağı oluşturduğu, işlettiği ve yönettiği, dolayısıyla yalnızca kooperatife ve güvene bağlı olan geçici ağların ortaya çıkmasından bu yana, bu son yaklaşımlar sınıfı özellikle önemli hale gelmiştir. Dahası, ilgili cihazların sınırlı kapasitesi hem önemli depolama hem de hesaplama gereksinimlerini zorunlu kılar. Böyle bir ağ genellikle acil bir talebi ve belirli bir hedefi karşılamak için oluşturulur ve düğümler sürekli olarak ona katılır veya ayrılır. Bu nedenle, dağıtılmış ve iş birliğine dayalı şemalara dayalı grup anahtarı yönetiminin büyük ilgi gördüğü kanıtlanmıştır (Merwe et all 2007).

Dağıtık ortamda en çok alıntı yapılan yaklaşımlardan biri (Steiner et all 1996) de ele alınmıştır. Bu çalışmalarda yazarlar, Diffie and Hellman (1976) geleneksel

Diffie-Hellman anahtar değişimini genişleten ve çok verimli yeniden anahtarlama prosedürleri içeren iki farklı grup anahtar yönetim şeması sağlar.

Yukarıda bahsedilen klasik Diffie-Hellman anahtar değişimini keyfi grup etkisine genellemektedir (Juan Antonio et all 2015).

Protokol 4.1.1. (Yarıgrup Diffie – Hellman Anahtar Değişimi)

S sonlu bir küme, G bir değişmeli yarı grup ve $\phi: G \times S \to S$, S üzerinde bir G etkisi olsun. (G, S, ϕ) içindeki yarı grup Diffie-Hellman anahtar değişimi protokolü aşağıda gibidir:

1- Alice ve Bob ortak bir $s \in S$ elemanı üzerinde anlaşırlar.
2- Alice bir $a \in G$ seçer ve $\phi(a, s)$'yi hesaplar. Alice'nin gizli anahtarı a, açık anahtarı $\phi(a, s)$ olur.
3- Bob bir $b \in G$ seçer ve $\phi(b, s)$'yi hesaplar. Bob'un gizli anahtarı b, açık anahtarı $\phi(b, s)$ olur.
4- Daha sonra onların ortak gizli anahtarı
$\Phi(a, \Phi(b, s)) = \Phi(ab, s) = \Phi(ba, s) = \Phi(b, \Phi(a, s))$ olur.

Orijinal Diffie-Hellman önerisinde, eğer bir düşman sözde Ayrık Logaritma Problemini (ALP) çözebilirse, o zaman Diffie-Hellman anahtar değişimini kırabilir. Bu ortamda, aşağıdaki daha genel sorunu benzer şekilde ele alabiliriz (Juan Antonio et all 2015).

Problem 4.1.2. (Yarıgrup Etki Problemi, YEP): Bir S kümesine ve $x, y \in S$ elemanlarına etki eden bir yarıgrup G verildiğinde, $\Phi(g, x) = y$ olacak şekilde g $\in G$ 'yi bulun.

Açıktır ki, eğer bir düşman olan Eve, $\Phi(g, s) = \Phi(a, s)$ olacak şekilde $g \in G$ bulursa, o zaman paylaşılan sırrı hesaplayarak $\Phi(g, \Phi(b, s)) = \Phi(gb, s) =$

$\Phi\,(bg, s) = \Phi\,(b, \Phi\,(a, s))$ olarak bulabilir. Önceki protokolün güvenliğinin aşağıdaki problemle eşdeğer olduğunu söyleyebiliriz (Juan Antonio et all 2015).

Problem 4.1.3. (Diffie – Hellman Yarıgrup Etki Problemi, DHYEP): Sonlu bir S kümesine ve bazı $g, h \in G$ için $y = \Phi\,(g, x)$ ve $z = \Phi\,(h, x)$'ye sahip $x, y, z \in S$ elemanlarına etki eden sonlu bir değişmeli yarı grup G verildiğinde $\Phi\,(gh, x)$'i bulun.

Yukarıda belirtildiği gibi, SAP'nin çözülmesi DHYEP'ın çözülmesini gerektirse de, her iki sorunun da (genel olarak) eşdeğer olup olmadığını bilmiyoruz, geleneksel Diffie – Hellman gibi kurulan, ancak burada bazı denklik sonuçları belirli senaryolar olduğu bilinmektedir (Juan Antonio et all 2015).

Yukarıdakilerden motive olarak, şimdi fikrimiz yarı grup Diffie – Hellman anahtar değişim protokolünün uzantılarını n kullanıcıya genelleştirerek ve sonra orijinal protokole kıyasla daha uygun özelliklere sahip diğer ayarları göz önünde bulundurarak tanımlamaktır.

Cihazların kapasitesi genellikle sınırlı olduğundan ve kimlik doğrulama işlemlerinin dağıtılmış bir ağda uygulanması zor olabileceğinden, pasif saldırılar altında dikkatimizi gizliliğe odaklıyoruz.

Bazı standart olmayan ayarlar daha genel örnekler olarak sunulmuştur, ancak SAP'nin sertliği henüz kanıtlanmış olmayabilir, bu nedenle bu durumlarda protokollerin güvenliği buna bağlıdır (Juan Antonio et all 2015).

Bölüm 4.1. boyunca, sonlu bir S kümesinin gizli bir öğesini paylaşmak isteyen $U_1, \ldots, U_n$ adlı bir n kullanıcı grubunu ele alacağız ve G, S üzerinde hareket eden sonlu bir değişmeli yarı grubu göstereceğiz.

B) Tek taraflı etkilere göre grup anahtar iletişimi

Bu bölümde, yarı grup Diffie – Hellman anahtar değişiminin üç farklı uzantısını, farklı hesaplama gereksinimleri ve iletişim ek yükleri ile ancak farklı durumlarda olası uygulamalarla ele alıyoruz.

a) Sıralı bir anahtar anlaşması

Anahtar değişim protokolünü genişletmek için ilk yaklaşım, bir kullanıcı zinciri boyunca özel bilgiler kullanılarak oluşturulan bir mesaj dizisinden ve ters yönde

analog bir ikinci mesaj dizisinden oluşur. Bu nedenle, iletişimi başlatan ve mesaj dizisini alan son kullanıcı hariç her kullanıcı iki mesaj gönderip alacaktır. Protokol aşağıdaki adımlarla tanımlanır.

Protokol 4.1.4. (GSAP-1). Kullanıcılar sonlu bir S kümesindeki bir s elemanı, sonlu bir değişmeli yarı grup G ve Φ tarafından verilen S üzerinde bir G etkisi üzerinde anlaşırlar. $\forall\, i = 1, \dots, n$ için $\mathcal{U}_i$ kullanıcısı özel bir $g_i \in G$ ye sahiptir.

i. Her $i = 1,2, \dots . n-1$, için $\mathcal{U}_i$ kullanıcısı $\mathcal{U}_{i+1}$ kullanıcısına $\{C_1, \dots . C_i\} = \{\Phi(g_1, s), \Phi(g_2 g_1, s), \dots \Phi(\prod_{j=1}^{i} g_j, s)\}$ mesajını gönderir.

ii. $\mathcal{U}_n$ kullanıcısı $\Phi(g_n, C_{n-1})$ i hesaplar.

iii. $k = n, \dots ,2$, için $\mathcal{U}_k$ kullanıcısı $\mathcal{U}_{k-1}$ kullanıcısına $\{f_1^k, \dots . f_{k-1}^k\}$ mesajını gönderir. Burada $f_j^k = \Phi(g_k, f_j^{k+1})$ için $2 \le k \le n-1$ ve $\Phi(g_n, C_{j-1})$, $j = 1, \dots , n-1$ ile $C_0 = s$.

iv. $\mathcal{U}_k$ kullanıcısı $\Phi(g_k, f_k^{k+1})$ i hesaplar.

---

b) Yayında önemli bir anlaşma

Aşağıdaki protokol, GSAP-1'den daha düşük bir iletişim ek yükü sunar. Buradaki fikir yine kullanıcı $\mathcal{U}_1$ den $\mathcal{U}_n$ kullanıcısına ilk mesaj dizisini almaktır, ancak şimdi $\mathcal{U}_n$, diğer kullanıcıların ortak anahtarı kurtarmasına izin veren bir mesaj yayınlayacaktır.

Protokol 4.1.5. (GSAP-2): Kullanıcılar sonlu bir S kümesindeki bir s elemanı, sonlu bir değişmeli yarı grup G ve S üzerinde bir G -etkisi Φ konusunda anlaşırlar. $\forall\, i = 1, \dots, n$ için $\mathcal{U}_i$ kullanıcısı özel bir $g_i \in G$ ye sahiptir.

i. Her $i = 1,2, \dots . n-1$, için $\mathcal{U}_i$ kullanıcısı $\mathcal{U}_{i+1}$ kullanıcısına

$\{C_{i-1}^{i-1}, C_1^i, \ldots., C_i^i\}$ mesajını gönderir. Burada $C_0^0 = s$, $C_1^1 = \Phi(g_1, s)$ ve $i \geq 2$ için $C_1^i = \Phi\left(g_i, C_{i-2}^{i-2}\right)$, $C_j^i = \Phi\left(g_i, C_{j-1}^{i-1}\right)$(ile $j = 2, \ldots \ldots, i$).

ii. $\mathcal{U}_n$ kullanıcısı $\Phi(g_n, C_{n-1}^{n-1})$ i hesaplar.

iii. $\mathcal{U}_n$ kullanıcısı $\{f_1^n, \ldots \ldots f_{n-1}^n, f_n^n\}$ yayınlar, burada $i = 1,2, \ldots . n-1$ için $f_1^n = \Phi(g_n, C_{n-1-i}^{n-1})$ olmak üzere $f_{n-1}^n = \Phi(g_n, C_{n-2}^{n-2})$ ve $f_n^n = C_{n-1}^{n-1}$'dir.

iv. $\mathcal{U}_i$ kullanıcısı $\Phi(g_i, f_i^n)$ i hesaplar.

Açıklama 4.1.6. GSAP-2 Protokolünün 3. adımında yayın mesajında bulunan f_n^n elemanına, $\mathcal{U}_i$ $i = 1, \ldots, n-1$ kullanıcıları tarafından paylaşılan anahtarı kurtarmak için herhangi biri tarafından ihtiyaç duyulmadığı gözlemlenebilir. Bununla birlikte, daha sonra göstereceğimiz gibi, gelecekteki yeniden anahtarlama işlemleri için bu değerin dağıtılması gereklidir. (Juan Antonio et all 2015).

Örnek 4.1.7. Önceki iki protokol çarpımsal yarı grup N^*'ın $\Phi(y, g^x) = (g^x)^y$ tarafından verilen, g tarafından üretilen q sıralı döngüsel bir grup S üzerindeki etki için eklenenlerin uzantılarıdır. Esas olarak hem tur sayısı hem de gönderilecek mesajlar nedeniyle aşırı iletişim giderleri sunar. Bu nedenle, yalnızca IKA.1 olarak adlandırılan ikincisi önerilir. Bununla birlikte, ilk protokol, iletişimlerinin güvenli olması gereken ve her düğümün düzgün çalışıp çalışmadığının nerede değerlendirilmesi gereken bir sensör ağına uygulandığında kendi başına ilginç olabilir. $\mathcal{U}_n$ kullanıcısı 1. adımda mesajı aldıktan sonra, azalan tur zincirinde mesajlardan herhangi birinin olmaması (sonuncusu hariç), karşılık gelen gönderen düğümün çalışmadığını veya iletişimin kesildiğini bildirir.

Örnek 4.1.8. Özellikle, sonlu bir $GF(q)$ alanını ve asal mertebeden bir g elemanını düşünün. $x, y \in N$ için N^* yarı gurubu, $\langle g \rangle \subset GF(q)^*$ ile $\Phi(y, g^x) = (g^x)^y$ alt grubuna etki eder.

Örnek 4.1.9. ε eliptik bir eğrideki noktalar kümesi olmak üzere $\forall\, n \in N$ için $\Phi(n, P) = nP$ ile verilen $\Phi: N^* \times \varepsilon \longrightarrow \varepsilon$ etkisi $\forall\, p \in \varepsilon$ eliptik eğriler için önceki protokollerin karşılık gelen versiyonlarını sağlar.

c) Bir grup etkisine dayalı anahtar anlaşması

Yarı-grup G'de S kümesine etki eden terslerin varlığı, azaltılmış iletişim giderleri ile ortak bir anahtar üzerinde anlaşmanın bir yolunu sağlayabilir. Ayrıca, hesaplamalar kullanıcılar arasında daha eşit dağıtılabilir. Önceki iki bölümde verilen protokollerde, bu gereksinimler ne kadar yüksekse, kullanıcının protokolü başlatan kişiden o kadar uzakta olduğunu belirtiyoruz. Böylece, G'nin bir grup olduğunu varsayıyoruz. Protokol aşağıdaki adımlarla verilir:

Protokol 4.1.10. (GSAP-3): Kullanıcılar sonlu bir S kümesindeki bir $C_0 = s$ elemanı, sonlu bir değişmeli yarı grup G ve S üzerinde bir G -etkisi Φ konusunda anlaşırlar. $\forall\, i = 1, \dots, n$ için $\mathcal{U}_i$ kullanıcısı özel bir $g_i \in G$ ye sahiptir.

i. Her $i = 1,2, \dots . n-2$, için $\mathcal{U}_i$ kullanıcısı $\mathcal{U}_{i+1}$ kullanıcısına $C_i = \Phi(g_i, C_{i-1})$ mesajını gönderir.

ii. $\mathcal{U}_{n-1}$ kullanıcısı $\Phi(g_{n-1}, C_{n-1})$'i hesaplar ve diğer kullanıcılara $[\mathcal{U}_{1,} \dots . \mathcal{U}_{n-2,} \mathcal{U}_n]$ olarak gönderir.

iii. $\mathcal{U}_n$ kullanıcısı $\Phi(g_n, C_{n-1})$'i hesaplar.

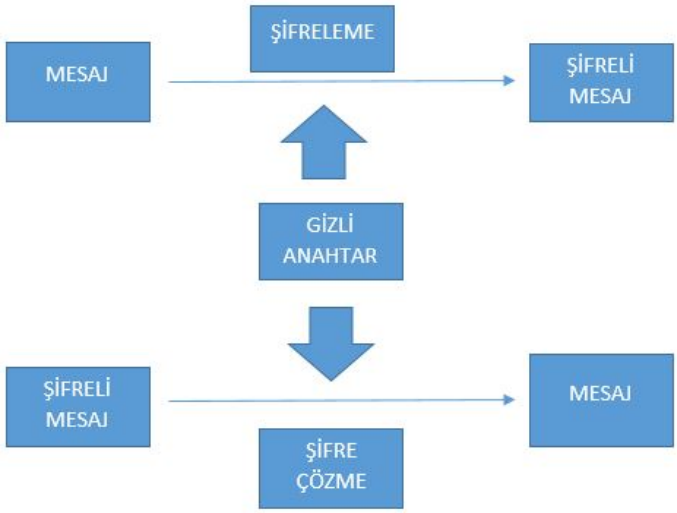
iv. $i = 1,2, \dots . n-1$ için $\mathcal{U}_i$ kullanıcısı $D_i = \Phi({g_i}^{-1}, C_{n-1})$ hesaplar ve $\mathcal{U}_n$ kullanıcısına gönderir.

v. $i = 1,2, \dots . n-1$ için $\mathcal{U}_n$ kullanıcısı $\Phi(g_n, D_{n-1})$ hesaplar ve $\{\mathcal{U}_{1,} \dots . \mathcal{U}_{n-2,} \mathcal{U}_{n-1}\}$ kullanıcılarına $\{\Phi(g_n, D_1), \dots . \Phi(g_n, D_{n-1}), C_{n-1}\}$ değerler kümesini gönderir.

vi. $i = 1,2, \dots . n-1$ için $\mathcal{U}_i$ kullanıcısı $\Phi\big(g_i, \Phi(g_n, D_i)\big)$'i hesaplar.

4.2. Grup Etkisi Üzerine Yapılan Anahtar Anlaşması Protokolleri

Alice ve Bob'un birbirleriyle şifreli olarak iletişim kurabilmeleri için Şekil 4.2.1'deki gibi gizli bir anahtara ihtiyaç duyduklarını varsayıyoruz.



Şekil 4.2.1. Alice ve Bob'un iletişim şeması

Anahtar anlaşması protokolü için saf bir örnek, bir tarafın gizli bir anahtarı yazması, onu kurcalamaya açık bir zarfa yerleştirmesi ve alıcıya göndermesidir. Zarf sağlamsa, gizli anahtar mesajları şifrelemek ve şifresini çözmek için her iki taraf tarafından da kullanılabilir. Şimdi anahtar anlaşma şemasını göstermeye çalışalım. Gizli anahtar anlaşması için, Alice ve Bob aşağıda verilen Tablo 1'deki kendi adımlarını takip etsin. (Çağman vd 2020).

S.N.	ALİCE	BOB
1	Rastgele bir n sayısı seçer ve herkese açık olarak Bob'a gönderir.	
2	α sıralı n^2'lusunu alır.	β sıralı n^2'lusunu alır.
3	$\forall$ pozitif $k \leq n^2$ tamsayısı için $i = \left\|\frac{k-1}{n}\right\|+1$, $j = \big((k-1)modn\big)+1$ ve $M_{ij} = \begin{pmatrix} cos\alpha_k & -sin\alpha_k \\ sin\alpha_k & cos\alpha_k \end{pmatrix} \in SO_2$ o.ü. nxn tipinde olan $M=\begin{bmatrix} M_{11} & \cdots & M_{1n} \\ \vdots & \ddots & \vdots \\ M_{n1} & \cdots & M_{nn} \end{bmatrix}$ matrisini oluşturur.	$\forall$ pozitif $k \leq n^2$ tamsayısı için $i = \left\|\frac{k-1}{n}\right\|+1$, $j = \big((k-1)modn\big)+1$ ve $N_{ij} = \begin{pmatrix} cos\beta_k & -sin\beta_k \\ sin\beta_k & cos\beta \end{pmatrix} \in SO_2$ o.ü. mxm tipinde olan $N=\begin{bmatrix} N_{11} & \cdots & N_{1n} \\ \vdots & \ddots & \vdots \\ N_{n1} & \cdots & N_{nn} \end{bmatrix}$ matrisini oluşturur.
4	Rastgele bir $z \in \mathbb{P}^1_{\mathbb{C}}$ komplex sayısı seçilir.	
5	$\forall$ pozitif $k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, $j = \big((k-1)modn\big)+1$ o.ü. $w_k = M_{ij} \otimes z$ ile $w = (w_1, w_2 \ldots . w_{n^2})$ sıralı n^2lisini hesaplar ve açık olarak Bob'a gönderir.	$\forall$ pozitif $k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, j=$((k-1)modn)$+1 o.ü. v_k=$N_{ij} \otimes z$ ile v=$(v, v_2 \ldots . v_{n^2})$ sıralı n^2lisini hesaplar ve açık olarak Alice gönderir.
6	Alice Bob tarafından gönderilen $v = (v_1, v_2 \ldots . v_{n^2})$ ile birlikte $\forall$ pozitif $k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, $j = \big((k-1)modn\big)+1$ o.ü. $p_k = M_{ij} \otimes v_k$ ile $\otimes$ işlemin sonucunda gizli anahtar olan $p = (p_1, p_2 \ldots . p_{n^2})$ sıralı n^2lisini bulur.	Bob Alice tarafından gönderilen $w = (w_1, w_2 \ldots . w_{n^2})$ ile birlikte $\forall$ pozitif $k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, $j = \big((k-1)modn\big)+1$ o.ü. $q_k = N_{ij} \otimes w_k$ ile $\otimes$ işlemin sonucunda gizli anahtar olan $q = (q_1, q_2 \ldots . q_{n^2})$ sıralı n^2lisini bulur.
7	$(p_1, p_2 \ldots . p_{n^2})$'lerin reel kısımlarının yan yana yazılmasıyla oluşan ortak anahtar olan K bulunur.	$(q_1, q_2 \ldots . q_{n^2})$'lerin reel kısımlarının yan yana yazılmasıyla oluşan ortak anahtar olan K bulunur.

Tablo 4.2.1. Anahtar Anlaşma Şeması

Anahtar anlaşma şemasının adımlarını kısaca açıklarsak;

İlk 4 adımda, Alice ve Bob değişkenlerine açık veya gizli olarak kullanılmak üzere belirli değerler atar.

Adım 5, Örnek 2.1'deki $\otimes$ işlemini kullanarak diğer tarafa gönderilecek olan w_k ve v_k için etki işlemini gerçekleştirir

Adım 6, her $k \leq n^2$ için $p_k = q_k$ olur. Bu da p'nin q ile aynı karmaşık sayılar listesine sahip olduğu anlamına gelir.

Aslında, $\forall\, k \leq n^2$ için;

$$\begin{aligned} p_k &= M_{ij} \otimes v_k \\ &= M_{ij} \otimes \left(N_{ij} \otimes v_k\right) (\otimes\text{'nın uyumluluk koşulu}) \\ &= \left(M_{ij}N_{ij}\right) \otimes z \\ &= \left(N_{ij}M_{ij}\right) \otimes z \ (\mathrm{SO}_2(\mathrm{R}) \text{ değişmeli}) \\ &= N_{ij} \otimes \left(M_{ij} \otimes z\right) (\otimes\text{'nın uyumluluk koşulu}) \\ &= N_{ij} \otimes w_k \\ &= q_k \end{aligned}$$

olur.

Adım 7 olmadan, karmaşık sayıların listesi p (= q) gizli anahtar olarak kullanılabilir, ancak daha basit bir gizli anahtar biçimi elde etmek için, bu listeler son adımda aynı tam sayı olarak K'ye dönüştürülür.

Oscar'ın sırasıyla Alice ve Bob'un özel anahtarlarından olan M ve N'den en az birini elde etmek isteğini varsayalım.

Bu amaca ulaşmak için z, w (veya v) karmaşık sayılarını kullanarak M (veya N) 'yi bulmalıdır, bu da her biri bilinmeyenlerden daha az denklem içeren n^2li doğrusal denklem sistemlerini çözmesi gerektiği anlamına gelir.

Bu doğrusal denklem sistemlerinin sonsuz sayıda çözümü olduğundan, anahtarı elde etmek hesaplama açısından çok zor olacaktır.

Mathematica Script Language yardımıyla bir uygulamadaki anahtar anlaşma şemasındaki bu adımları inceleyelim.

Örnek 4.2.2. Alice, n sayısını 3 olarak ayarlar ve herkese açık olarak Bob'a gönderir. Sonra, Alice ve Bob sırasıyla n^2li

$$\alpha = (703.58,540.02,212.64,567.73,579.75,299.43,957.68,43.095,379.52)$$

ve

$$\beta = (389.19,456.71,157.93,999.24,278.39,181.05,586.63,295.77,372.74)$$

sıralı 9'lusunu alır.

$$M = \begin{pmatrix} 0.990664 & 0.136329 & 0.944759 & 0.327765 & 0.550110 & 0.835092 \\ -0.136329 & 0.990664 & -0.327765 & 0.944759 & -0.835092 & 0.55011 \\ -0.622964 & -0.782251 & -0.125821 & -0.992053 & -0.558103 & 0.829771 \\ 0.782251 & -0.622964 & 0.992053 & -0.125821 & -0.829771 & -0.558103 \\ -0.874807 & -0.484472 & 0.63151 & 0.775368 & -0.818092 & -0.575087 \\ 0.484472 & -0.874807 & -0.775368 & 0.63151 & 0.575087 & -0.818092 \end{pmatrix}$$

$$N = \begin{pmatrix} 0.933232 & 0.359273 & -0.381789 & 0.92425 & 0.659707 & -0.751523 \\ -0.359273 & 0.933232 & -0.92425 & -0.381789 & -0.751523 & 0.659707 \\ 0.977288 & -0.211917 & -0.351385 & -0.936231 & 0.397162 & 0.917749 \\ 0.211917 & 0.977288 & 0.936231 & -0.351385 & -0.917749 & 0.397162 \\ -0.661615 & -0.749844 & 0.895923 & -0.444208 & -0.445086 & -0.895488 \\ 0.749844 & -0.661615 & -0.444208 & 0.895923 & 0.895488 & -0.445086 \end{pmatrix}$$

Ayrıca, z karmaşık sayısının 14.75 + 125.36i olarak belirlendiğini varsayalım.

Bir sonraki adımda;

Alice $\forall$ pozitif $k \leq n^2$, $i = \left\| \frac{k-1}{n} \right\| + 1$, $j = \big((k-1) mod n\big) + 1$, $w_k = M_{ij} \otimes z$ olmak üzere $w = (w_1, w_2 \ldots . w_{n^2})$ sıralı n^2lisini hesaplar ve açık olarak Bob'a gönderir.

$$w = (-7.292270 + 0.427684i, -2.889400 + 0.073594i, -0.660011 + 0.0112959i,$$

$$-0.797807 + 0.0128766i, -0.127762 + 0.00799653i, 0.671195 + 0.011413i,$$

$$-1.80916 + 0.0336278i, -0.815922 + 0.0131067i, -1.42509 + 0.0238502i)$$

Bob ∀ pozitif $k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, $j = \big((k-1)modn\big) + 1$, $v_k = N_{ij} \otimes z$ olmak üzere $v = (v_1, v_2 \ldots . v_{n^2})$ sıralı n^2lisini hesaplar ve açık olarak Alice gönderir.

$v = (-2.60349 + 0.061225i, 0.411967 + 0.00920353i, 0.876093 +$

$0.0139078i, 4.58489 + 0.173489i, -0.376349 + 0.0089826i, -0.433825 +$

$0.00934901i, -0.883887 + 0.0140158i, 2.01159 + 0.0397161i, -0.498149$

$+0.00982073i)$

Sonra Alice Bob tarafından gönderilen $v = (v_1, v_2 \ldots . v_{n^2})$ ile birlikte ∀ pozitif

$k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, $j = \big((k-1)modn\big) + 1$, $p_k = M_{ij} \otimes v_k$ olmak üzere $\otimes$ işlemin sonucunda gizli anahtar olan $p = (p_1, p_2 \ldots . p_{n^2})$ sıralı n^2lisini bulur.

$p = (-1.81566 + 0.033813i, 0.885394 + 0.0140368i, -7.22916 + 0.420425i,$

$-1.22683 + 0.0197124i, 1.89186 + 0.0360372i, -5.40077 + 0.237798i,$

$-0.221649 + 0.00825466i, -2.2024 + 0.0460466i, 0.0151652 + 0.0080491i)$

Diğer taraftan Bob Alice tarafından gönderilen $w = (w_1, w_2 \ldots . w_{n^2})$ ile birlikte ∀ pozitif $k \leq n^2$, $i = \left\|\frac{k-1}{n}\right\| + 1$, $j = \big((k-1)modn\big) + 1$, $q_k = N_{ij} \otimes w_k$ olmak üzere $\otimes$ işlemin sonucunda gizli anahtar olan $q = (q_1, q_2 \ldots . q_{n^2})$ sıralı n^2lisini bulur.

$q = (-1.81566 + 0.033813i, 0.885394 + 0.0140368i, -7.22916 + 0.420425i,$

$-1.22683 + 0.0197124i, 1.89186 + 0.0360372i, -5.40077 + 0.237798i,$

$-0.221649 + 0.00825466i, -2.2024 + 0.0460466i, 0.0151652 + 0.0080491i)$

Son olarak, p sıralı n^2lilerinin gerçel ve sanal kısımlarının birleştirilmesi ile oluşturulan ve ayni şekilde q sıralı n^2lilerinin gerçel ve sanal kısımlarının birleştirilmesi ile oluşturulan sayılar aynıdır ve oluşturulan K gizli anahtar olur.

$K = 1815660033813088539400140368722916042042512268300197124189$

$1860036037254007702377980221649000825466220240046046601516520008049$1 (Çağman vd 2020).

5. SONUÇLAR

İnsanoğlunun varoluşundan bu yana haberleşme çok önemli bir konudur. Günümüzde Ülkeler arası yaşanılan anlaşmazlık ve savaşlar yüzünden haberleşme sistemlerinin güvenliği büyük önem arz etmektedir. Haberleşme işleminde güvenliğin ve gizliliğin sağlanması için kullanılan en önemli işlem, gönderilen mesajın şifrelenerek anlamsız hale getirilip alıcıya gönderilmesi ve alıcı mesaja tersi işlem yaparak yani deşifrelenerek tekrar eski haline getirilmesidir. Bu yapılan işlemler Anahtar anlaşmasının bir parçasıdır.

Biz bu çalışmada Kriptanaliz, kriptografi ve kriptoloji hakkında bilgiler verip kriptolojinin tarihçesi üzerinde durduk, ilerleyen bölümde anahtar anlaşması daha sonra yarı grup ve grup etkisi üzerine yapılan anahtar anlaşması protokollerinin nasıl olması gerektiğini açıkladık.

Yapılan bu çalışma kişiler, devletler kısacası haberleşme durumunu kullanan herkes için haberleşmede güvenlik zafiyetinin oluşmaması için yapılması gerekenler hakkında bilgilendirme verilmiştir. Anahtar anlaşması ile ilgili Türkçe olarak yapılan çok fazla çalışma olmadığından dolayı çalışmamız bizden sonra yapılacak olan çalışmalar için kaynak oluşturup onlara yol gösterecektir.

KAYNAKLAR

Altan, K., Kaşkaloğlu, K., Kındap, N., Özakın, Ç., Saygı, Z., Yıldırım, E., Yıldırım, M. ve Yıldız, S., Akış Şifreler, 2004. Kriptolojiye Giriş Ders Notları. ODTÜ Yayıncılık, Ankara, 32,33.

Anshel, I., Anshel, M. and Goldfeld D., 1999. An algebraic method for public-key cryptography. Mathematical Research Letters, 3–4, 287–291.

Çağman, A., Polat K. and Taş S., 2020. A key agreement protocol based on group actions. Numerical Methods for Partial Differential Equations, 3-8.

Caroline, J. K., 2006. Special Signature schemes and key agreement protocols. Doktora tezi, Londra Üniversitesi.

Çimen, C., Akleylek, S. ve Akyıldız, E., 2007. Şifrelerin matematiği kriptografi. ODTÜ Yayıncılık, Ankara, 5-10.

Demirel, M., 2019. Kriptoloji ve eski şifreleme yönetimleri. https://medium.com/@msuhademirel/kriptografi-ve-eski-%C5%9Fifreleme-y%C3%B6ntemleri-ancient-cryptography-techniques-e64311de1629 (29.03.2021).

Diffie, W. and Hellman, M. E., 1976. New directions in cryptography, IEEE Trans. Inform. Theory IT-22, 6, 644–654.

Kindap, N., 2015. Kriptolojinin tarihçesi. Cryptography Timeline, https://tr.linkedin.com/pulse/kriptolojinin-tarih%C3%A7esi-nihal-kindap (29.03.2021).

Schnyder, R., López-ramos, J., Rosenthal, J. and Schipani, D; 2016. Group key management based on semigroup actions. World Scientific Publishing Co Pte Ltd, 1-6

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

Stinson R. D. and Paterson B. M., 2019. Cryptography theory and practice. CRC Press, 600 p, New York, London.

https://tr.wikipedia.org/wiki/Diffie-Hellman_anahtar_değişimi (29.03.2021).

https://www.britannica.com/topic/cryptology/History-of-cryptology

https://tr.wikipedia.org/wiki/Kriptoloji

https://www.geeksforgeeks.org/classical-cryptography-and-quantum-cryptography/#:~:text=In%20the%20classical%20cryptography%20the,plain%20text%20to%20cipher%20text.

https://en.wikipedia.org/wiki/Multiple_encryption

https://www.wolfssl.com/what-is-a-block-cipher/#:~:text=A%20block%20cipher%20is%20an,%2C%20192%2C%20or%20256%20bits

https://tr.wikipedia.org/wiki/RSA_(%C5%9Fifreleme_y%C3%B6netimi)

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Ahmet ÇÖP
Doğum Yeri ve Tarihi	
Eğitim Durumu	
Lisans Öğrenimi	Erzincan Üniversitesi Fen Edebiyat Fakültesi Matematik Bölümü
Yüksek Lisans Öğrenimi	Ağrı İbrahim Çeçen Üniversitesi Lisansüstü Eğitim Enstitüsü Matematik Anabilim Dalı
İş Deneyimi	
Çalıştığı Kurumlar	Ordu Açı Koleji, Ağrı İl Emniyet Müdürlüğü
İletişim	
E-posta Adresi	