LİNEER KRİPTANALİZ

1. DENEYİN AMACI

Bu deney, simetrik şifreleme algoritması kullanılarak şifrelenmiş bir metnin, üçüncü bir kişi (düşman) tarafından lineer kriptanaliz yöntemi ile nasıl çözülebileceğini göstermeyi amaçlamaktadır. Şifre kırma yöntemi olarak gösterilecek olan lineer kriptanaliz, [1] dökümanında detaylı olarak açıklanmıştır. Burada amaç lineer kriptanalizi öğretmek olduğundan, üzerinde kriptanaliz yapılabilecek basitleştirilmiş bir şifreleme algoritması seçilmiştir.

2. DES (Data Encryption Standard) ALGORİTMASI

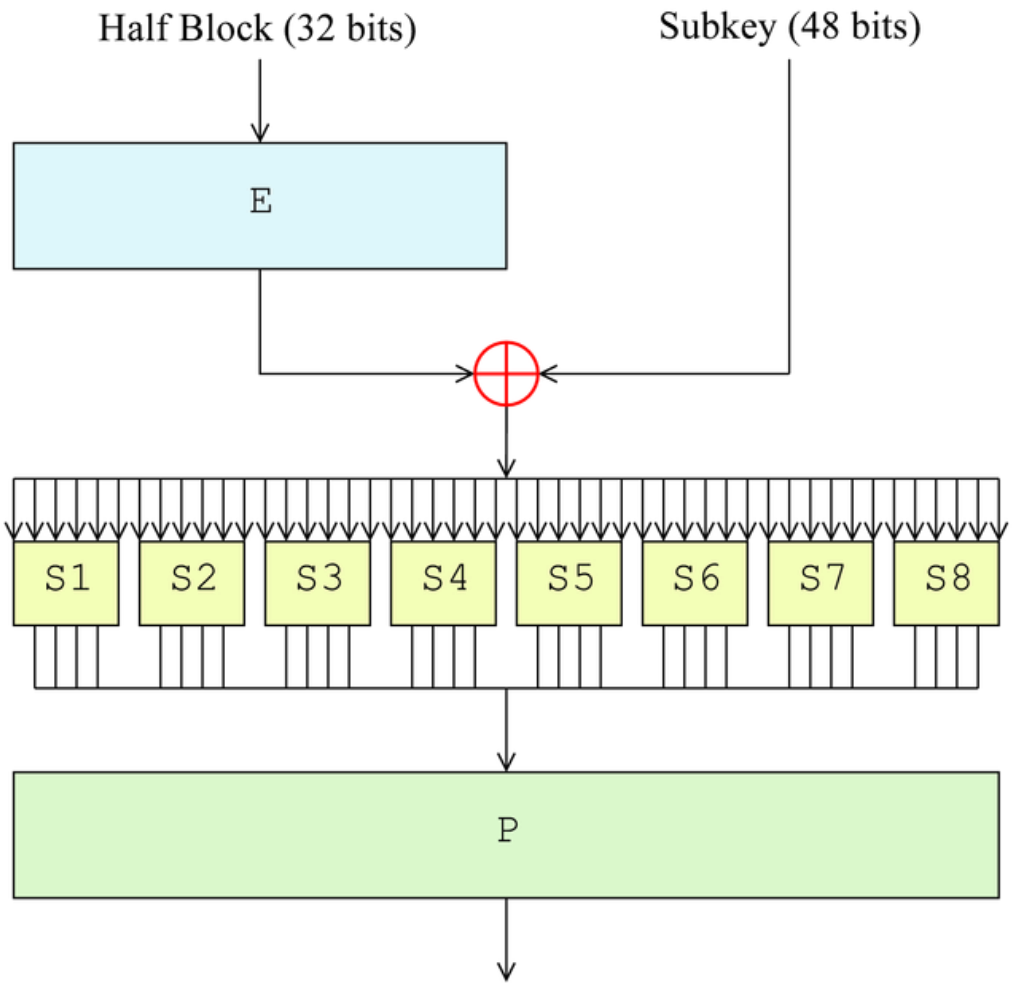
Açılımı Data Encryption Standart olan simetrik şifreleme algoritmasıdır. 1997'de resmi bilgi şifreleme standardı olarak kabul edilirken, 2000'de yerini AES'e bırakmıştır.

DES, gizli anahtarlı bir şifreleme türüdür, büyük boyutlu verilerin şifrelenmesinde kullanılır. Şifreleme işlemi Blok Şifreleme olarak adlandırılan bir yöntem ile gerçekleştirilir. Bu yöntem, şifreli metin ile düz metin arasındaki ilişkiyi gizlemeyi amaçlar. Her şifreleme adımına döngü denilir ve her döngüde kullanılan anahtar farklıdır. Açık mesaj, belirli uzunluktaki bloklara bölünür ve ayrı ayrı şifrelenen bloklar ile şifreli metin elde edilir. Her bir blok, 8 bit parity biti olmak suretiyle, 64 bit uzunluğundadır. Blok uzunluğu, kullanılan işlemci hızına göre değişebilir. Yeni dönem bilgisayarlarda, 128 bit kullanılmaya başlanmıştır.

DES şifrelemenin en büyük dezavantajı yavaş olmasıdır. Bu yöntemde bilinmezlik fazladır; her bir bloğun her biti, diğer bitler ve anahtar ile bağımlıdır.

DES, bağımlılık fazla olmasına rağmen, modern bilgisayarlara dayanamaz. Brute Force ataklarına karşı güvensizdir. Bu noktada DES'in güvenilirliğini artırmak için 3DES yöntemi geliştirilmiştir. Bu yöntemde, şifrelenen veri farklı bir anahtar ile tekrar geri çözülür ve DES şifrelemesi 3 sefer ard arda yapılır. Şifreleme için kullanılan ve uzunluğu 24 byte olan anahtar, 3 bloğa ayrılır. İlk 8 byte ile şifreleme yapılır, buraya kadar olan kısım DES işlemidir. Daha sonra şifrelenen metin ortadaki 8 byte ile çözülür ve son 8 byte ile tekrar şifrelenerek 8 byte'lık blok elde edilir. DES'e göre güvenilirliği fazladır fakat hız 3 kat daha azalmıştır. Her byte bir eşlik biti bulundurur. Dolayısı ile kullanılan anahtar 168 bittir. (24*7=168)

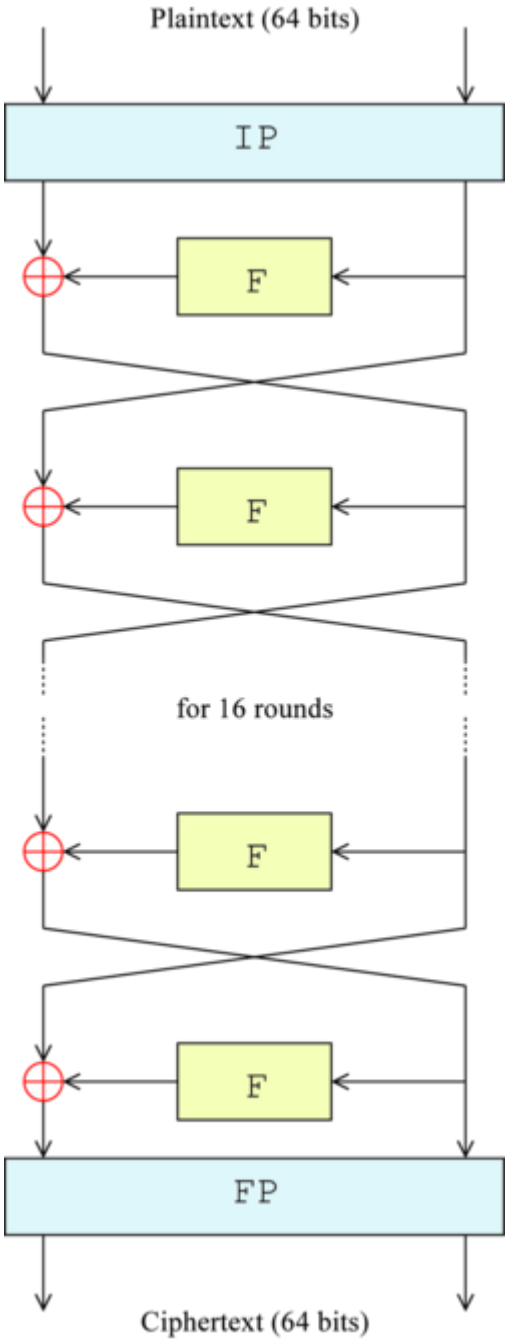
DES'i kırmak için yüksek maliyetle son teknoloji makineler geliştirilmiş olmasına rağmen 3DES, bankalar ve devlet daireleri olmak üzere birçok ortamda kullanılmaya devam etmektedir.



Şekil 1. Feistel (F) Fonksiyonu

DES Algoritması, Feistel fonksiyonlarının ardışık kullanımından oluşmaktadır. Şekil 1 de Feistel algoritması, Şekil 2 de de DES Algoritması gösterilmiştir. Burada amaç, DES algortimasını detaylı olarak anlatmak yerine, DES Algoritmasının genel yapısını anlatmaktır.

DES i güvenli kılan kısmı, lineer olmayan bir transform işlemi olan S-box fonksiyonudur. Şekil 1 de 8 adet S-box (S1 – S8) görülmektedir. Her bir S-box, 6 bitlik datayı alarak 4 bitlik data üretmektedir. S-box fonksiyonunun doğruluk tablosu Şekil 3 de gösterilmiştir.



Şekil 2. DES Algoritması

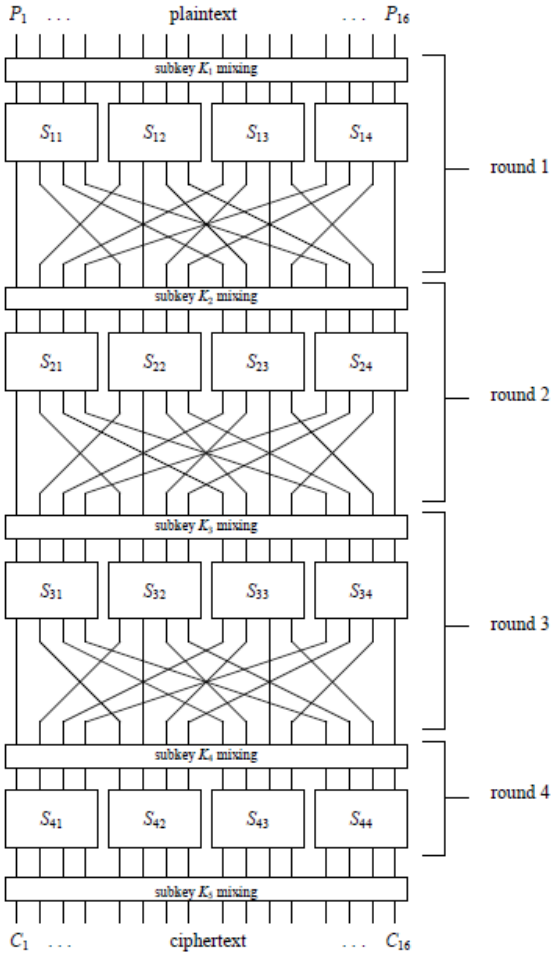
S_5		Orta 4 bit															
		0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**
İlk ve son bitler	**00**	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

Şekil 3. 6x4 bitlik S-box

Örnek olarak, S-box fonksiyonuna input olarak 011011 verirsek, ilk ve son bit 01 olacak, orta 4 bit de 1101 olacaktır. Sonuç, 1001 olur.

3. BASİT ŞİFRELEME YÖNTEMİ

Kullanacağımız şifreleme yöntemi, DES e benzer şekilde dizayn edilmiş, temel Yerine Koyma-Permütasyon Ağı fikrine dayalı bir yöntemdir. Bu yöntem, Şekil 4 de gösterilmiştir. Şekilde 4 roundluk bir algoritma gösterilmiş olmakla beraber, biz lineer kriptanaliz için 2 roundluk bir algoritma kullanacağız.



Şekil 4. Kullanacağımız Basit şifreleme yönteminin 4-round hali.

Her döngü (round), 3 işlemden oluşmaktadır:

- Yerine Koyma: Burada yerine koyma fonksiyonu, şekilde S harfi ile gösterilmiştir. Yerine koyma işlemi, Tablo 1 de gösterilmiştir.

input	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
output	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7

Tablo 1. Yerine Koyma Fonksiyonu (hexadecimal olarak gösterilmiştir).

- Permütasyon: Permütasyon işlemi, eldeki verinin bitlerinin basit bir şekilde yer değiştirmesidir. Kullandığımız şifreleme algoritmasındaki permütasyon işlemi, Tablo 2 de gösterilmiştir. Örnek olarak, 10. sıradaki bitin yeri değiştirilerek 7. sıraya yazılacaktır.

input	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
output	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16

Tablo 2. Permütasyon Fonksiyonu (hexadecimal olarak gösterilmiştir).

- Anahtar Ekleme: Anahtar ekleme işlemi, basit bir XOR işlemidir. Round anahtarı, şekil 4 de gösterildiği gibi, permütasyon işleminden geçmiş data ile XOR lanacaktır.

Tersine şifreleme işlemi, şifreleme algoritmasında uygulanan işlemlerin tam tersi uygulanarak yapılır.

4. LİNEER KRİPTANALİZ

Lineer kriptanaliz; düz metin, şifreli metin ve döngü anahtarlarının bitleri kullanılarak oluşturulan lineer denklemlerin bazılarının yüksek olasılıkla doğru olmasını kullanılarak yapılan atak şeklidir. Atak yapan kişinin elinde düz metin-şifreli metin çiftlerinin olması durumunda yapılabilen bir ataktır. Ancak, atak yapan kişi, istediği düz metin-şifreli metin çiftini oluşturamaz. Dolayısıyla bu çiftlerin rastgele bilgiler olduğu varsayılır.

Buradaki temel fikir şudur: şifreleme yapılırken kullanılan herhangi bir denklemi, lineer olan başka bir denklem şeklinde yazmaktır. Bahsedilen bu lineer denklem şu şekildedir:

$$X_{i_1} \oplus X_{i_2} \oplus \ldots \oplus X_{i_u} \oplus Y_{j_1} \oplus Y_{j_2} \oplus \ldots \oplus Y_{j_v} = 0$$

Burada X_i, işlemin girdisi olan X in i. bitini, Y_j de işlemin çıktısı olan Y nin j. bitini ifade etmektedir. Bu eşitlik, u girdi bitinin ve v çıktı bitinin birbiriyle XOR lanmasını göstermektedir.

Lineer kriptanalizde amaç, yukarıda açıklanan biçimde denklemler yazarak, bu denklemlerin yüksek veya düşük olasılıkla doğru olanlarını belirlemektir. Eğer bir şifreleme yönteminde yukarıdaki şekilde olan ve yüksek veya düşük olasılıkla doğru olan denklemler bulabilirsek, bu bize bu şifreleme yönteminin zayıflığını gösterir. Mükemmel bir şifreleme algoritması olsaydı,

yukarıdaki şekilde yazılabilecek herhangi bir denklemin doğru olma olasılığı ½ olurdu. Lineer kriptanaliz, ½ olasılığından sapmaları inceleyerek bu sapmaları kullanmaya çalışmaktadır.

Yığma Prensibi

Elimizde iki rastgele ikili değişken olduğunu varsayalım, bunlar da X_1 ve X_2 olsun. $X_1 \oplus X_2 = 0$ lineer denkleminin doğruluk tablosunu oluşturalım. Burada, X_1 ve X_2 nin olasılık dağılımı aşağıdaki gibi ise:

$$\Pr(X_1 = i) = \begin{cases} p_1 & ,i = 0 \\ 1 - p_1 & ,i = 1 \end{cases}$$

$$\Pr(X_2 = i) = \begin{cases} p_2 & ,i = 0 \\ 1 - p_2 & ,i = 1. \end{cases}$$

ve iki değişken birbirinden bağımsız ise:

$$\Pr(X_1 = i, X_2 = j) = \begin{cases} p_1 p_2 & ,i = 0, j = 0 \\ p_1(1 - p_2) & ,i = 0, j = 1 \\ (1 - p_1)p_2 & ,i = 1, j = 0 \\ (1 - p_1)(1 - p_2) & ,i = 1, j = 1 \end{cases}$$

ve buradan şunu çıkarabiliriz:

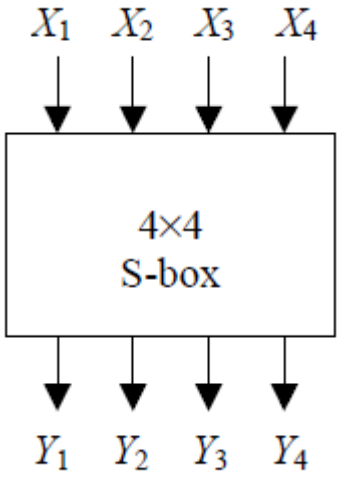
$$\begin{aligned} \Pr(X_1 \oplus X_2 = 0) &= \Pr(X_1 = X_2) \\ &= \Pr(X_1 = 0, X_2 = 0) + \Pr(X_1 = 1, X_2 = 1) \\ &= p_1 p_2 + (1 - p_1)(1 - p_2). \end{aligned}$$

Burada $p_1 = ½ + \varepsilon_1$ ve $p_2 = ½ + \varepsilon_2$ şeklinde yazarsak,

$$\Pr(X_1 \oplus X_2 = 0) = 1/2 + 2\varepsilon_1\varepsilon_2$$

olur.

S-Box Analizi



$$X_2 \oplus X_3 = Y_1 \oplus Y_3 \oplus Y_4 .$$

Lineer denklemini, X in bütün olası değerleri için incelersek, 16 durumdan 12 sinde doğru olduğunu görürüz. Yani bu denklemin doğruluk oranı 12/16 = ½ + ¼ olur.

X ve Y değerlerinin olası bütün lineer denklem kombinasyon denklemlerini incelersek, aşağıdaki tabloyu elde ederiz.

		Output Sum															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Input Sum	0	+8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	−2	−2	0	0	−2	+6	+2	+2	0	0	+2	+2	0	0
	2	0	0	−2	−2	0	0	−2	−2	0	0	+2	+2	0	0	−6	+2
	3	0	0	0	0	0	0	0	0	+2	−6	−2	−2	+2	+2	−2	−2
	4	0	+2	0	−2	−2	−4	−2	0	0	−2	0	+2	+2	−4	+2	0
	5	0	−2	−2	0	−2	0	+4	+2	−2	0	−4	+2	0	−2	−2	0
	6	0	+2	−2	+4	+2	0	0	+2	0	−2	+2	+4	−2	0	0	−2
	7	0	−2	0	+2	+2	−4	+2	0	−2	0	+2	0	+4	+2	0	+2
	8	0	0	0	0	0	0	0	0	−2	+2	+2	−2	+2	−2	−2	−6
	9	0	0	−2	−2	0	0	−2	−2	−4	0	−2	+2	0	+4	+2	−2
	A	0	+4	−2	+2	−4	0	+2	−2	+2	+2	0	0	+2	+2	0	0
	B	0	+4	0	−4	+4	0	+4	0	0	0	0	0	0	0	0	0
	C	0	−2	+4	−2	−2	0	+2	0	+2	0	+2	+4	0	+2	0	−2
	D	0	+2	+2	0	−2	+4	0	+2	−4	−2	+2	0	+2	0	0	+2
	E	0	+2	+2	0	−2	−4	0	+2	−2	0	0	−2	−4	+2	−2	0
	F	0	−2	−4	−2	−2	0	+2	0	0	−2	+4	−2	−2	0	+2	0

Table 4. Linear Approximation Table

Bu deneyde sizden istenen, yukarıdaki lineer tahmin tablosunu hesaplayıp ekrana basmanızdır.