DÜZENSİZ ŞİFRELEME ALGORİTMASININ GERÇEK ZAMANLI KRİPTO ANALİZİ

Esen Akkemik[†‡], **Orhun Kara**[†]
[†] TÜBİTAK UEKAE
[‡]ODTÜ UME Kriptografi
{esena, orhun}@uekae.tubitak.gov.tr

ABSTRACT

In this work, we have cryptanalyzed the "Unsystematic Cipher" designed in the last decade. The *Unsystematic Cipher* is an example of a stream cipher. It is claimed by the designers that the "Unsystematic Cipher" is even more advantageous, in all aspects than One Time Pad, and hence the cipher is proposed to substitute for One Time Pad.

We have observed that the *Unsystematic Cipher* suffers from key diffusion drastically. We have exploited the poor diffusion property to mount two divide and conquer type attacks on the cipher: a ciphertext-only attack and a known plaintext attack. Both attacks have ignorable time complexities. One can implement the attacks even by hand on a paper without using any computer. Moreover, both attacks require few data. For example, it is possible to recover the key by only 4-5 known plaintexts or 10-12 ciphertexts. It is very rare to encounter such a successful attack in terms of complexity in cryptology literature, except the ones mounted on historical ciphers. Thus, it is surprising that a modern cipher designed by some scientists and published in a scientific journal is extremely weak.

Key words: Stream cipher, unsystematic cipher, cryptanalysis, known-plaintext attack, ciphertext-only attack, divide-and-conquer attack.

1. GİRİŞ

Tek-Kullanımlık-Istampa (One-Time-Pad) 1917 yılında Vernam tarafından tasarlanmış bir şifreleme sistemidir [3]. Açık metnin bit sayısı kadar uzunluğunda tamamen rastgele bir anahtar dizisi ile açık metin bitlerinin d-ya'sına (dışarlayıcı ya, exor) dayanır. Açık metin P, anahtar dizisi K, açık metin bit sayısı N ise şifreli metin olan C'nin bitleri şu şekilde belirlenir: $C_i=P_i \oplus K_i$, $i=1,...,N$. Bu sistem mükemmel gizliliği sağlar [3], yani Sadece Şifreli Metin saldırısı uygulamak sonsuz hesapsal güç sahibi olunsa dahi imkansızdır. Yalnız, mükemmel gizliliği sağlamak için anahtar dizisinin sadece bir kere kullanılması şarttır.

Düzensiz Şifreleme Algoritması Taş, Alataş ve Akın tarafından ELECO2002 sempozyumunda sunulmuş [2], 2004 yılında da İstanbul Üniversitesi Elektrik-Elektronik Mühendisliği dergisinde yayınlanmış bir akış (stream) şifreleme tekniğidir [1]. Tasarımcıları, algoritmanın anahtar dizisini tekrar kullanabilmesinden dolayı Tek-Kullanımlık-Istampaya karşı üstünlük sağladığını iddia etmişlerdir [1, 2].

Bu makalede *Düzensiz Şifreleme* Algoritması analiz edilmiş ve algoritmaya uygulanan iki atak örneği anlatılmıştır. Bu ataklar, algoritmaya geliştirilmiş ilk ataklardır. Ayrıca ataklar karmaşıklıkları açısından değerlendirildiklerinde son derece uygulanabilir türdendir.

Algoritmada anahtar yayınımı (key diffusion) açısından ciddi sorunlar olduğu gözlenmiştir. Anahtar yayınımındaki zayıflık algoritmaya "böl-ve-fethet" türünden atak geliştirilmesinde kullanılmıştır. Algoritmaya hem Bilinen Açık Metin Atağ, hem de Sadece Şifreli Metin Atağı düzenlenmiştir. Her iki atağın da zaman karmaşıklığı yok denecek kadar azdır. Hatta, bir bilgisayara dahi ihtiyaç duymadan atakları kağıt üzerinde gerçeklemek mümkündür. Ayrıca atakların veri karmaşıklıkları da son derece düşüktür. Bilinen Açık Metin Atağını uygulamak için 4-5 açık metin yeterli olmaktadır. Sadece Şifreli Metin Atağında ise yaklaşık 10-12 şifreli metin ile anahtar ele geçirilebilmektedir. Bu sonuçlar *Düzensiz Şifreleme* Algoritmasının ne kadar güvensiz bir algoritma olduğunu göstermektedir.

Bu makalede Bölüm 2'de *Düzensiz Şifreleme* Algoritması kısaca anlatılmıştır. Şifreleme sistemlerinde kullanılan bazı atak çeşitleri ve algoritmaya geliştirilen ataklar Bölüm 3'te verilmiştir. Ayrıca, Ek bölümünde her iki atak için birer örnek sunulmuştur.

2. DÜZENSİZ ŞİFRELEME ALGORİTMASI

Düzensiz Şifreleme Algoritması bir akış şifreleme tekniğidir. Her seferinde bitler bazında şifreleme yapılır. Öncelikle açık metin karakterlerinin kaç

bitle (n) simgeleneceği belirlenir. Bu değer belirlendikten sonra $\{1,...,n\}$ aralığından rastgele bir k sayısı seçilir. Bu k değerine göre açık metin bitleri k gruba ayrılır. Bu ayırma işleminde her grupta mümkün olduğunca eşit eleman olmasına dikkat edilir. Eşit elemanın olmadığı durumlarda da fazla eleman taşıyan gruplar, önemli bitlerin olduğu tarafta olacak şekilde gruplandırma yapılır. Daha sonra rastgele k bit (IV) üretilir. Bu bitlerin değeri 1 ise açık metnin gruplanmış halinde karşılık gelen grup bitleri değiştirilir, aksi takdirde aynen alınır. Bu işlem sonunda elde edilen n bitin arkasına k bitlik rastgele değer eklenerek n bitlik blok için şifreli metinde karşılık gelen $n+k$ bitlik blok elde edilir. Aynı işlemler bir sonraki karakter bloğu için tekrarlanır. Burada anahtar rastgele seçilen bitlerin (IV) uzunluğudur. Şifreleme metodu bir örnek üzerinde aşağıdaki gibidir:

Açık metin ($P = P_1, P_2$): 01000101 01000001
Anahtar (k_1 , k_2): {4, 3 }
IV (IV_1, IV_2): 1101 011
Gruplama: 01 00 01 01 010 000 01
1 1 0 1 0 1 1
Şifreleme: 10 11 01 10 010 111 10
Şifreli metin ($C=C_1,C_2$):

10110110110101011110011 (1)

Bu çalışmada rastgele üretilen ve şifreli metinde açık şekilde yollanan rastsal değerler başlangıç vektörü (IV) olarak adlandırılmıştır. Şifreleme işlemlerinde karakterlerin ASCII gösteriminin kullanıldığı, yani her açık metin karakterinin en fazla 8 bitle simgelendiği varsayılmıştır.

3. ŞİFRELEME SİSTEMİNİN KRİPTO ANALİZİ

Bu makalede *Düzensiz Şifreleme* Algoritmasına bir Sadece Şifreli Metin Atağı ve bir de Bilinen Açık Metin Atağı yapılmıştır. Bu atak türlerinin kısa açıklaması aşağıdaki gibidir:

1. Bilinen Açık Metin Atağı: Atak yapanın elinde bir grup açık metin ve karşılık gelen şifreli metinler vardır. Bu veriler kullanılarak anahtar ele geçirilir. 1994 yılında Matsui tarafından bulunan doğrusal kriptoanaliz bu tür bir ataktır [4]. Bu atak ile DES 2^{47} bilinen açık metin bloğu (bir blok 64 bit uzunluğundadır) kullanılarak kırılmıştır.
2. Sadece Şifreli Metin Atağı: Atak yapan kişi, elindeki şifreli metinleri kullanarak anahtarı veya açık metni bulmaya çalışır. Matsui'nin DES'e yaptığı Doğrusal Ataklardan birisi de bu türdendir. Atağın veri karmaşıklığı yaklaşık 2^{54} şifreli metin bloğudur [4].

Her bir karakteri şifrelemek için 3 bit entropisi olan anahtar parçası kullanılmaktadır. Dolayısıyla, N karakter uzunluğunda açık metni şifrelemek için 3N uzunluğunda anahtar bilgisi kullanılır. Anahtar değerinin tek tek denenmesine dayanan kaba kuvvet atağının karmaşıklığı bu algoritma için 2^{3N} olur. Örneğin, 100 karakterlik açık metne karşılık gelen şifreli metinde kaba kuvvet atağının karmaşıklığı 2^{300}'dür. 4GHz hızında bir bilgisayar saniyede 2^{32} anahtar tararsa, 2^{300} anahtarı 2^{268} saniyede dener. Bu işlem bir milyar bilgisayarda yaklaşık $1.5x\ 10^{64}$ yıl sürer. Ancak, aşağıda verilen her iki atak yöntemi de kaba kuvvet atağıyla kıyaslanamayacak kadar düşük karmaşıklığa sahip olduklaarı için gerçek zamanda uygulanabilirdir.

3.1 Sadece Şifreli Metin Analizi

Şifreleme işleminde İngilizce alfabenin kullanıldığı varsayılsın. Bu durumda bütün İngilizce karakterlerin, rakamların ve noktalama işaretlerinin bitsel gösterimi en fazla 7 bit ile yapılır. Eğer şifreleme işleminde her karakter 8 bit ile gösterilirse her karakterin en önemli biti 0 olur. Bu ayırt edici özellik yardımıyla aşağıdaki önerme geçerlidir.

Önerme 1: Şifreli metinde bir karakter (8 bit) açık metindeki bir karakterin şifreli hali olduğu zaman şifreli metindeki karakterin en önemli biti ile IV'nin en önemli bitinin d-ya'sı açık metnin en önemli bitine eşittir.

İspat: Açık metnin en önemli biti 0 olsun. IV uzunluğu $\{1,...,8\}$ aralığında herhangi bir değer olsun. IV'nin en önemli biti 1 ise açık metnin en önemli biti değiştirileceğinden dolayı şifreli metinde karşılık gelen karakterde en önemli bit 1 olacaktır. IV'nin en önemli biti 0 olursa açık metnin en önemli biti değiştirilmeyeceği için şifreli metinde en önemli bit 0 olacaktır. Açık metnin en önemli bitinin bir olması durumu benzerdir. □

Sadece İngilizce alfabenin kullanıldığı açık metinlerde Önerme 1 şifreli metin karakterinin en önemli bitinin IV'nin en önemli bitine eşit olması gerektiğini söyler.

Aşağıda anlatılan "böl-ve-yönet" tekniğindeki atak, yani her seferinde şifreli metnin bir karakteri ile işlemler yapıp bir sonraki karakterine geçen atak, bu ayırt edici özellik ve Önerme 1 kullanılarak geliştirilmiştir.

Elde S tane şifreli metin olsun. Amaç, sadece şifreli metinler kullanarak "böl-ve-yönet" tekniği ile anahtar dizisini elde etmektir. Atağın $i$inci adımı şu şekildedir:

$(i-1)$inci adımdaki anahtar $k_{i-1}=m$ olsun. Bundan önce de şifreli metinlerde t bit ilerlenmiş olsun. $C_{(t+1)}...C_{(t+8)}$ bitleri açık metin karakterinin

şifrelenmiş hali olarak varsayılıp $C_{(t+9)}$ bitinin $C_{(t+1)}$ bitine eşit olup olmadığına bakılır, yani Önerme 1'in geçerliliği kontrol edilir. Bu değerler eşitse Önerme 1 sağlanıyor demektir ve önceki anahtar değerleri doğru varsayılıp k_i=1 kabul edilir ve $C_{(t+10)}$ ...$C_{(t+17)}$ bitleri bir sonraki açık metin karakterinin şifrelenmiş hali olarak alınır. Eğer $C_{(t+9)}$ biti $C_{(t+1)}$ bitine eşit değilse $k_{i-1}=m+1$ olarak kabul edilir. Bu durumda $C_{(t+2)}$...$C_{(t+9)}$ bitleri açık metnin şifrelenmiş hali olarak farz edilip $C_{(t+10)}$ bitinin $C_{(t+2)}$ bitine eşit olup olmadığına bakılır. Bu değerler eşitse k_i=1 kabul edilir ve $C_{(t+11)}$...$C_{(t+18)}$ bitleri bir sonraki açık metnin şifrelenmiş hali olarak kabul edilir. Atak bu şekilde devam ettirilir. *C* ile simgelenen şifreli metin, eldeki bütün şifreli metinlerin her birini ifade eder. Anahtar değeri belirleme işlemi yapılırken şifreli metin bitlerinin kıyaslanmasında eşit olmama durumunun en az bir şifreli metinde, eşit olma durumunun da bütün şifreli metinlerde sağlanması gereklidir.

Bu atak algoritma olarak şu şekilde verilir:
d dizisi açık metin karakterinin olası şifreli halini ve IV'nin en önemli bit değerini tutan 9 elemanlı dizi, *m* değeri {1,...,8} aralığında bir değer olsun. *m* değeri 8'i aştığı zaman algoritma sonlandırılmalıdır. *t*'nin ilk değeri 0 alınarak atağa başlanır.

$k_{i-1} = m$; $j = 0$;
$d = \{ C_{t+1}, C_{t+2}, ..., C_{t+8}, C_{t+9} \}$;
eğer ($d[1] \neq d[9]$) ise {
    ($d[1] \neq d[9]$) iken {
        $k_{i-1} = k_{i-1}+1$; $j = j+1$;
        $d = \{ C_{t+1+j}, ..., C_{t+8+j}, C_{t+9+j} \}$;}
}
eğer ($d[1] = d[9]$) ise {
    $k_i = 1$;
    $d = \{ C_{t+10+j}, ..., C_{t+17+j}, C_{t+18+j} \}$;
}

Bu atağın başarısı, şifre çözme boyunca anahtar için doğru tahminin yapılmasına bağlıdır. Atakta bit kıyaslama işleminde her şifreli metin için herhangi bir adımda Önerme 1 sağlanıyorsa 0, sağlanmıyorsa 1 olacak şekilde bir vektör oluşturulsun. Bir karakter şifre çözme işleminde 8 farklı anahtar değeri için 8 vektör elde edilir. Doğru anahtar için bu vektör 0 vektörü olacaktır. Ancak, yanlış anahtar değeri için de sıfır olabilir. Bu durumu sağlayan anahtar değerleri atakta yanlış alarma, yani anahtar değeri yanlış iken doğru kabul etmeye, sebep olur. Bu durumda elde olan *S* tane şifreli metin için herhangi bir adımdaki yanlış alarm olasılığı

$$P_f = 1 - (1 - 2^{-S})^7 \qquad (2)$$

olarak hesaplanır.
Bir karakter için birden fazla 0 vektör olması birden fazla anahtarın şifre çözme işleminde kullanılabileceğinin göstergesidir. Ancak, bu durum sonraki karakterlerin şifre çözümünde uyumsuzluk yaratıp hatayı tespit etme imkanı oluşturabileceği gibi yapılan bir hata başka hata/hatalarla birleşip doğru karakter/IV çiftine tekrar ulaşabilir. Böyle bir durumda, bir adımdan sonra bir grup anahtar yanlış tahmin edilmiş ancak sonraki bir adımda tekrar doğru anahtar değerlerine ulaşılmış demektir.

Tablo 1. Sadece Şifreli Metin atağında şifreli metin sayısına (*S*) karşılık atağın yanlış alarm olasılığı (P_f)

S	P_f
5	0,2
6	0,1
7	0,05
8	0,027
9	0,014
10	0,0068
20	$6{,}6 \times 10^{-6}$
30	$6{,}5 \times 10^{-9}$

Denklem 2 ve Tablo 1 değerleriyle elde edilen şifreli metin sayısı arttıkça atakla tahmin edilen anahtar dizisinin doğru olma olasılığının arttığı sonucuna varılır. Tablo 1'den de görüldüğü gibi yaklaşık 20 şifreli metin, anahtarın neredeyse %100 olasılıkla ele geçirilmesi için yeterli olmaktadır.

3.2 Bilinen Açık Metin Analizi

Bu bölümde *Düzensiz Şifreleme* Algoritmasına bir Bilinen Açık Metin Atağı uygulaması anlatılmaktadır. Bu atakta çok daha az veri ile anahtarı bulmak mümkün olabilmektedir.

Atakta açık metin bitlerinin değiştirilip değiştirilmeyeceğine karar veren IV bitleri kullanılmaktadır. Atağın sömürdüğü ayırt edici özellik Önerme 2'de verilmektedir. Tablo 2'de herhangi bir açık metin karakterinin *i*inci biti şifrelenirken anahtar değerine bağlı olarak IV'nin kaçıncı bitinin kullanıldığı gösterilmektedir. Örneğin, anahtar değeri 1 iken her bir *i* değeri için k_i=1 olmaktadır.

Tablo 2. Anahtar değerine göre açık metnin şifrelemesinde IV'nin kullanılacak bitlerinin (k_i) gösterimi.

i	1	2	3	4	5	6	7	8
K=1, k_i	1	1	1	1	1	1	1	1
K=2, k_i	1	1	1	1	2	2	2	2
K=3, k_i	1	1	1	2	2	2	3	3
K=4, k_i	1	1	2	2	3	3	4	4
K=5, k_i	1	1	2	2	3	3	4	5
K=6, k_i	1	1	2	2	3	4	5	6
K=7, k_i	1	1	2	3	4	5	6	7
K=8, k_i	1	2	3	4	5	6	7	8

Önerme 2: Herhangi açık metin karakteri, karşılık gelen şifreli metin karakteri ve IV'nin bitleri arasında

$$P_i \oplus C_i \oplus IV_{k_i} = 0,\ i=1,...,8 \qquad (3)$$

eşitliği vardır.

İspat: IV_{k_i} değeri açık metin karakterinin $i$inci bitinin değiştirilip değiştirilmeyeceğini belirler. $IV_{k_i} = 0$ iken, açık metin bitleri değiştirilmeyeceği için karşılık gelen şifreli metin bitlerine eşit olacaktır. $IV_{k_i} = 1$ iken ise açık metin karakterindeki $i$inci bit 1 ile d-ya'lanacaktır. IV değeri de 1 olduğundan, Denklem 3 bu durumda da sağlanır. □

Dikkat edilecek olursa Önerme 2, Önerme 1'in genelleştirilmiş halidir. Buna rağmen, Önerme 1, Sadece Şifreli Metin Analizinde kullanıldığı için ayrıca vurgulanmıştır.

Bilinen Açık Metin Atağında, tıpkı Sadece Şifreli Metin atağında olduğu gibi her bir karakter için elde bulunan verilerden aday anahtar değerleri belirlenir. Bu durumda ayırt edici özellik olarak Önerme 1 yerine Önerme 2 kullanılır. Önerme 2'de verilen eşitlik şartını bütün veriler için sağlayan anahtar değerleri aday olarak belirlenir. Diğer taraftan, sadece birkaç tane açık-şifreli metin çifti bile aday olarak sadece gerçek anahtarın kalmasına yeterli olmaktadır.

Algoritma kodu aşağıda verilmiştir. P ile açık metin, d ile olası şifreli metin, IV ile şifrelemede kullanılacak olası IV, m ile açık metin karakter sayısı simgelenmektedir. k_{top} bulunulan adımdan önceki bütün anahtar değerlerinin toplamını gösterir. Bir dizinin sonuna yeni eleman ekleme | ile gösterilmiştir.

$k_{top} = 0$; $t = 0, ..., m - 1$ {

$d = \{ C_{8t + k_{top} + 1}, ..., C_{8t + k_{top} + 8} \}$

$P = P_{8t+1}, ..., P_{8t+8}$;

$k_{t+1} = 1$; ind $= 8t + 8 + k_{top} + 1$;

IV $= C_{\text{ind}}$;

$i = 1, ..., 8$ {

$d_i \oplus P_i \oplus IV_{k_i} \neq 0$ ise{

$k_{t+1} = k_{t+1} + 1$;

ind = ind + 1 ;

IV = IV | C_{ind} ;

$d_i \oplus P_i \oplus IV_{k_i}$ hesapla;

}

$t = t + 1$; $k_{\text{top}} = k_{\text{top}} + k_t$;

}

Bu atağın başarısı, şifre çözme boyunca anahtar için doğru tahminin yapılmasına bağlıdır. Elde S tane açık-şifreli metin çifti olsun. Bu durumda her bir anahtar adayının Denklem 3 ile verilen kontrol mekanizmasından geçme ihtimali 2^{-8S} olmaktadır. Dolayısıyla, belirli bir açık metin karakterini şifreleyen anahtar değeri dışında en az bir tane yanlış anahtar gelme ihtimali

$$P_f = (1 - (1 - 2^{-8S})^7) \qquad (4)$$

ifadesi ile verilir.

Tablo 3. Bilinen Açık Metin saldırısında açık metin/şifreli metin çifti sayısına (S) karşılık atağın yanlış alarm olasılığı (P_f)

S	P_f
5	6.4×10^{-12}
6	2.5×10^{-14}
7	9.7×10^{-17}
8	3.8×10^{-19}
9	$1{,}5 \times 10^{-21}$
10	$5{,}8 \times 10^{-24}$

Tablo 3'teki örneklerden de görüleceği gibi atağın yanlış alarm olasılığı son derece düşüktür. Öyle ki, birkaç tane açık metin-şifreli metin çifti nerdeyse bir olasılıkla anahtarı tespit etmek için yeterli olmaktadır.

Bu atakta da bir önceki atakta olduğu gibi yanlış anahtar değeri, ilerde tespit edilebilecek hataya neden olabileceği gibi sonradan kendini toparlayan bir duruma da sebep olabilir.

4. SONUÇ

Bu çalışmada *Düzensiz Şifreleme* Algoritmasına biri Sadece Şifreli Metin Atağı, diğeri Bilinen Açık Metin Atağı olmak üzere iki atak anlatılmıştır. Bu ataklar algoritmaya uygulanan ilk ataklardır ve atakların karmaşıklıkları son derece düşüktür. Sonuç olarak, *Düzensiz Şifreleme* Algoritmasının son derece zayıf bir algoritma olduğu anlaşılmaktadır.

KAYNAKLAR

[1] Oğuzhan Taş, Bilal Alataş, Erhan Akın, *A New approach to Stream Cipher: Unsystematic Cipher*, İstanbul University Journal of Electrical & Electronics Engineering (IU-JEEE), Issue 7, Year: 2004, Vol. 4, Number:1, pp. 1057-1062

[2] Oğuzhan Taş, Bilal Alataş, Erhan Akın , Akış Şifreleme Tekniğine Yeni Bir Yaklaşım: Düzensiz Şifreleme, Elektrik, Elektronik ve Bilgisayar Mühendisliği Sempozyumu ve Fuarı (ELECO2002), Aralık 2002, Bursa, Türkiye.

[3] Alfred J. Menezes, Paul C. Van Oorschot, Scott A. Vanstone, *Handbook of Applied Cryprography*, sayfa 42, CRC yayınları, 1. basım, Ekim 1996, ABD.

[4] Mitsuru Matsui, *Linear cryptanalysis method for DES cipher*, Advances in Cryptology, Proceedings of EUROCRYPT'93, LNCS 765, sayfa. 386-397, Springer-Verlag, 1993.

EK A

Sadece Şifreli Metin Atağı Uygulaması

Elde aynı anahtar dizisi ile şifrelenmiş 3 tane şifreli metin olsun. Açık metnin ilk bitinin 0 olduğu (Ingilizce metin şifrelenmesi) da verilsin. Bu durumda Önerme 1'de açık metin biti yerine 0 değeri konularak işlem yapılacaktır.

C_1: 01001000011101110010000010010
C_2: 11010110110110010101101000110
C_3: 10110101100001111000110110 1001

IV_i ile *i*inci IV değeri simgelensin. Öncelikle, şifreli metinlerin ilk 8 biti ayrılır. Önerme 1'den dolayı her zaman 9. bit şifreli metnin en önemli bitine (1. bit) eşittir.

C_1: 010010000 11101110 **0**100000010010
C_2: 110101101 10110010 **1**011010000110
C_3: 101101011 00001111 **0**001101101001

Atağa k_1=1 kabul edilerek başlanır. 9. bitten itibaren ikinci 8 bit şifreli metin bloğu olarak alınır. Eğer k_1=1 ise 18. bit IV'nin ilk biti olur. Ancak, C_1'in ilk biti ile IV_2 biribirine eşit değildirö yani Önerme 1 sağlanmaz. Bu durum bir çelişkidir, bu çelişki de k_1=1 varsayımından kaynaklanmıştır. Öyleyse, k_1=2 alınır ve işleme devam edilir. Böylelikle, aşağıdaki durum geçerlidir.

C_1: 0100100**01** 11011100 **1**00000010010
C_2: 1101011**11** 01100101 **0**11010000110
C_3: 1011010**110** 00011110 **00**1101101001

k_1=2 ve k_2=1 olursa 19. bit IV_2'nin ilk biti olur. Bu durumda, bütün şifreli metinlerin 11. bitleri ve IV_2 değeri Önerme 1'i sağlar. Böylelikle, bitler aşağıdaki gibi olur:

C_1: 0100100**01** 110111001 00000010 **010**
C_2: 1101011**11** 011001010 11010000 **110**
C_3: 1011010**110** 000111100 01101101 **001**

20. bitten itibaren 8 bit alındığında 28. bit IV_3'ün ilk bitidir. Bütün şifreli metinlerin 20. bitleri ve IV_3 değerleri Önerme 1'i sağlar. Öyleyse k_1=2 ve k_2=1 değerleri doğru tahmin edilmiş varsayılır ve şifreli metinde en son değerlendirmeye katılmayan bitler en son IV değeri olarak alındıktan sonra anahtar dizisi {2,1,3} olarak bulunur ve koyu yazılmış bitler de IV değerlerini belirtir.

C_1: 0100100**01** 110111001 00000010**010**
C_2: 1101011**11** 011001010 11010000**110**
C_3: 1011010**110** 000111100 01101101**001**

Bilinen Açık Metin Atağı Uygulaması

Açık metin *P* ve şifreli metin *C* ile gösterilsin.

P: 01000111 00100011 00011101 01100111
C: 010010000111011100100000100110101101 001101

Atağa şifreli metinde ilk 8 bit ayrılarak başlanır.

C: 01001000 **0**
11101110010000001001101011010011 01

k_1=1 varsayılır. Bu durumda IV_1=0 olur. Şifreli ve açık metnin 5. biti ile IV_1 Önerme 2'yi sağlamadıkları için k_1=2 alınır.Yeni IV_1=01 olur. Bu yeni IV, şifreli metnin ilk 8 biti ve açık metnin ilk karakteri Önerme 2'yi sağladığı için k_1=2 varsayılır.

P: 01000111 00100011 00011101 01100111
C: 0100100**01** 11011100 **1** 00000010010110111011001101

Şifreli metinden 11. bitten itibaren 8 bit alınır. k_2=1 alınır ve IV_2'nin değeri 19. bit (1) alınarak atağa başlanır. Açık metnin ikinci karakteri, şifreli metnin 11.,...,18. bitleri ve IV_2=1 Önerme 2'yi sağladığı için k_1=2 varsayımı hala geçerlidir ve k_2=1 alınarak üçüncü basamağa geçilir

C: 0100100**01** 110111001 00000010 **0** 110101101001101

Şifreli metinde 20. bitten itibaren 8 bit ayrılır. k_3=1 ise IV_3=0 olur. Açık metnin 3. karakterinde 4. bit, şifreli metnin 23. biti ve IV_3 Önerme 2'yi sağlamadığından dolayı k_3=2 yapılır. Yeni IV_3=01 alınır.

P: 01000111 00100011 00011101 01100111
C: 0100100**01** 110111001 00000010 **01** 10101101001101

Bu yeni IV değerine göre şifreli metnin 23. biti, açık metnin 3. karakterinin 4. biti ve IV_3'ün açık metnin bu bloğunun şifrelemesinde kullanılacak 1. biti Önerme 2'yi sağlamadığı için k_3=3 yapılır. Bu durumda IV_3=011 olur. Bu yeni IV_3, açık metnin üçüncü karakteri ve şifreli metnin 20.,...27. bitleri Önerme 2'yi sağladığı için k_3=3 kabul edilir.

P: 01000111 00100011 00011101 01100111
C: 0100100**01** 110111001 00000010**011** 0101101**01101**

Şifreli metinde 31. bitten itibaren 8 bit ayrılır. Geriye kalan 5 bit IV_4 değeridir. Açık metnin 4. karakteri, şifreli metnin 31,...38. bitleri ve IV_4 Önerme 2'yi sağladıından dolayı anahtar dizisi {2,1,3,5} olarak bulunur.