AES Blok Şifresinin Anahtar Genişletme Rutininin Geliştirilmesi ve Bir Blok Şifreden Bağımsız Anahtar Genişletme Rutininin Tasarımı

Fatma Büyüksaraçoğlu Sakallı, Ercan Buluş, Muharrem Tolga Sakallı, Hüseyin Vural

***Özet*—AES (Advanced Encryption Standard) blok şifresi 2001 yılında standart olmuş önemli bir simetrik şifreleme algoritmasıdır. Bununla beraber AES blok şifresinin saldırılara imkân tanıyan en önemli zaafı anahtar genişletme rutininin yavaş yayılım ve bit sızıntı problemlerine sahip olmasıdır. Bu çalışmada AES blok şifresinin anahtar genişletme rutinindeki bu problemleri gideren bir anahtar genişletme rutini geliştirilmektedir ve bu rutinden faydalanarak bir blok şifreden bağımsız anahtar genişletme rutininin nasıl tasarlanabileceği tartışılmaktadır.**

***Anahtar Kelimeler*—AES blok şifresi, blok şifreler, anahtar genişletme rutini, yeni bir anahtar genişletme mimarisi.**

***Abstract*—AES (Advanced Encryption Standard) block cipher, which has been deployed as a standard in 2001, is an important symmetric cipher. However, the key expansion routine of the AES has two important weaknesses, slow diffusion and bit leakage, which are used to execute some important attacks against AES. In the present study, a new improved key expansion routine for the AES eliminating these weaknesses is developed and how to design a key expansion routine independent from a block cipher is discussed.**

***Index Terms*— AES block cipher, block ciphers, key expansion routine, a new key expansion routine structure**

I. GİRİŞ

2001 yılında DES (Data Encryption Standard) [1] şifreleme algoritmasının yerini alan ve standart haline gelen AES (Advanced Encryption Standard) [2] blok şifresi 128-bit veri bloklarını 128-bit, 192-bit ve 256-bit anahtar seçenekleri ile şifreleyen bir blok şifreleme algoritmasıdır [1-3]. Döngü sayısı 128-bit, 192-bit ve 256-bit anahtar seçenekleri için sırasıyla 10, 12 ve 14 döngüdür. Her döngü dört adım içerir:

i) SubBytes (Byte Yerdeğiştirme),
ii) ShiftRows (Satırları Öteleme),
iii) MixColumns (Sütunları Karıştırma),
iv) AddRoundKey (Döngü Anahtarı Ekleme).

Her döngüde sırasıyla gerçekleştirilen bu adımlardan byte yer değiştirme adımında 8-bit (byte) değerleri farklı 8-bit (byte) değerleri ile yer değiştirilir (S-kutusu). Bu dönüşüm doğrusal olmayan bir dönüşümdür ve GF(2^8) sonlu cisminde ters haritalama tabanlıdır [2-5]. Satırları öteleme adımında byte değerlerinin permütasyonu ile byte değerlerinin sırası değiştirilirken, MixColumns doğrusal dönüşümde 32-bit giriş değerlerinden sabit bir matris çarpımı yardımıyla 32-bit çıkış değerleri elde edilmektedir. Diğer yandan son adım olan döngü anahtarı ekleme evresinde 128-bit anahtar seçeneği ile şifreleme yapan AES şifresi için anahtar genişletme evresinden gelen 128-bit anahtar değer ile o anki blok XOR'lama işlemine tabi tutulur. Şekil 1, tek döngülük SPN mimarisine uygun AES algoritmasını göstermektedir.

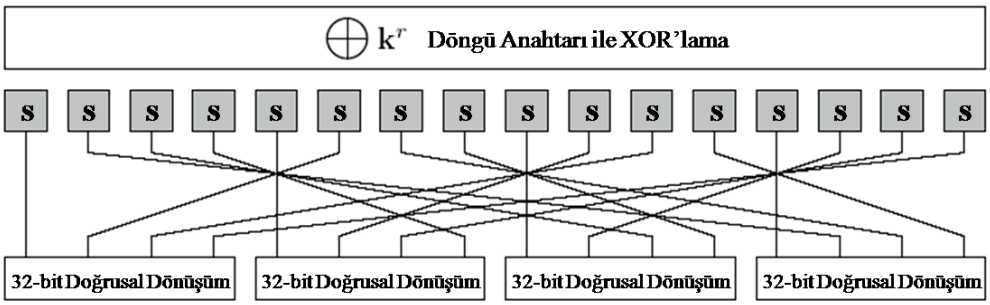
AES blok şifresinin anahtar genişletme algoritması, her ne kadar basit olsa da, bazı önemli saldırıların gerçekleştirilmesinde doğrudan etkisi olmasından dolayı döngü fonksiyonu kadar güçlü değildir. Bu saldırıların nedeni AES anahtar genişletme rutininin yavaş yayılım ve bit sızıntı problemlerine sahip olmasıdır. Yavaş yayılım problemi, AES-192 (192-bit anahtar kullanan AES blok şifresi) ve AES-256 (256-bit anahtar kullanan AES blok şifresi) için ilişkili anahtar saldırılarında kullanılmıştır [6-13]. Bit sızıntı probleminde ise saldırgan verilen bir alt anahtar bilgisi ile diğer alt anahtarları elde edilebilmektedir. Bu problem 7-döngülük AES-192 ve AES-256 için imkânsız diferansiyel saldırısında kullanılabilmektedir [14]. May vd. [15] AES blok şifresinde kullanılan anahtar genişletme rutininin güçlendirilmesi için AES döngü fonksiyonunun üç defa yürütüldüğü, her alt anahtarın birbirinden bağımsız olarak elde edildiği ve uygulama maliyeti yüksek yeni bir anahtar genişletme rutinini öne sürmüşlerdir. Bu çalışmada AES blok şifresinin anahtar genişletme rutininin orijinal yapısını bozmadan ve bahsedilen problemleri gideren yeni bir anahtar genişletme rutini önerilmektedir. Ayrıca önerilen bu yeni anahtar genişletme rutininden esinlenerek elde edilen blok şifreden bağımsız yeni bir anahtar genişletme mimarisi incelenmektedir.

Fatma Büyüksaraçoğlu Sakallı, Trakya Üniversitesi, Bilgisayar Mühendisliği Bölümü, Edirne, fbuyuksaracoglu@trakya.edu.tr

Ercan Buluş, Namık Kemal Üniversitesi, Bilgisayar Mühendisliği Bölümü, Çorlu-Tekirdağ, ercanbulus@nku.edu.tr

Muharrem Tolga SAKALLI, Trakya Üniversitesi, Bilgisayar Mühendisliği Bölümü, Edirne, tolga@ trakya.edu.tr

Hüseyin Vural, Trakya Üniversitesi, Bilgisayar Mühendisliği Bölümü, Edirne, huseyinvural02@gmail.com



Şekil 1. Tek döngülük AES algoritması

II. AES ANAHTAR GENİŞLETME RUTİNİ

Bu bölümde AES-128 için anahtar genişletme rutini incelenmektedir. Blok şifrenin AES-192 ve AES-256 iki versiyonu için de bazı küçük değişiklikler ile birlikte rutin AES-128'inki ile aynıdır. Şekil 2 de genel şekli verilen AES-128 anahtar genişletme rutini aşağıdaki gibi ifade edilebilir:

1- İlk 4 kelime (w_0, w_1, w_2, w_3) gizli anahtardan elde edilir. Gizli anahtar k_0 dan k_{15}'e kadar 16 byte bir dizi olarak düşünülür. İlk 4 byte (k_0 dan k_3'e) w_0, ikinci 4 byte (k_4'ten k_7'ye) w_1 ve benzer şekilde diğer kelimeler w_2 ve w_3'te gizli anahtarın kelimeler şeklinde yan yana konması ile elde edilir,

2- Diğer kelimeler w_i (i = 4 den 43'e kadar) aşağıdaki şekilde elde edilir:

   a. Eğer i (mod 4) $\neq 0$ ise $w_i = w_{i-1} \oplus w_{i-4}$ şeklinde tablodan da görüldüğü gibi soldan ve üstten bir değerden elde edilir.
   b. Eğer i (mod 4) = 0 ise $w_i = t \oplus w_{i-4}$ şeklinde elde edilir. Burada t geçici bir bellek ve iki rutinin w_{i-1} üzerindeki uygulama sonucudur: SubWord ve RotWord. t'nin elde edilme süreci bir döngü sabiti RCon ile XOR lama işlemi ile sonlanır.
   Diğer bir deyişle;
   $t = SubWord(RotWord(w_{i-1})) \oplus RCon_{i/4}$.

A. Kelime Döndürme (RotWord):

Bu rutin AES şifresinde kullanılan satırları öteleme (ShiftRows) dönüşümüne benzemektedir. Ancak sadece 1 satıra uygulanır. Bu rutin bir kelimeyi 4 bytelık bir dizisi olarak alır ve her byte'ı sola dairesel olarak öteler.

B. Kelime Yer değiştirme (SubWord):

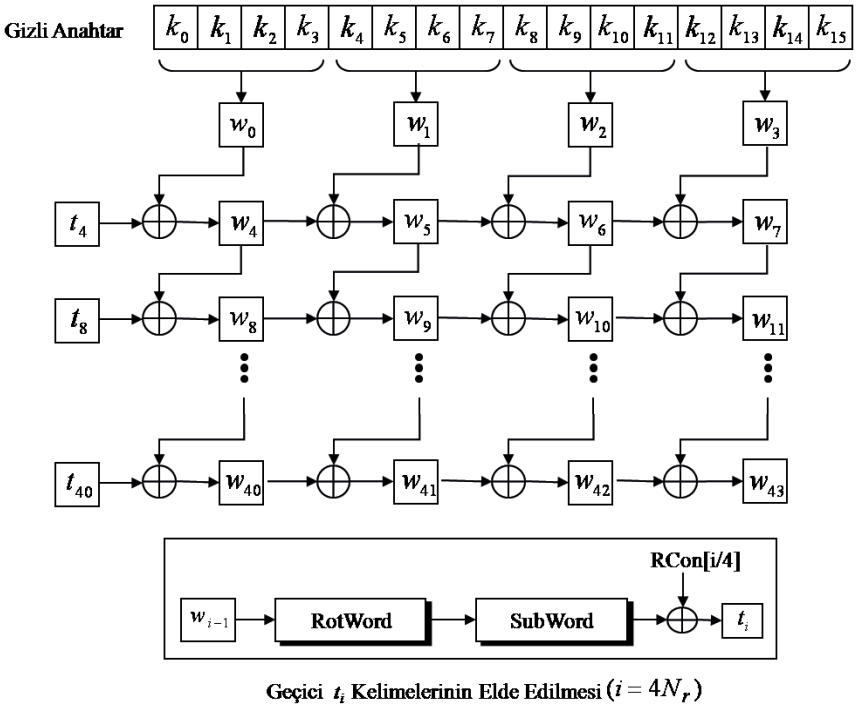
Bu rutin AES şifresinde kullanılan Byte yerdeğiştirme (SubBytes) dönüşümüne benzemektedir. Ancak sadece 4 byte'a uygulanır. Bu döngü kelimedeki her byte değerini alır ve diğer bir byte ile yerdeğiştirir.

C. Döngü Sabitleri (Round Constants):

Anahtar genişletme rutini her döngüde farklı sabit değer kullanır. Bu sabit, RCon, 4 byte değerinde ve en sağdaki 3 byte'ı 0 olan bir değerdir. AES-128 için (10 döngü için) farklı döngü sabitleri Tablo 1'de gösterilmektedir. Buna ek olarak Tablolarda kullanılan tüm ikili değerler hexadecimal (h alt simgesi ile gösterilmiştir) sayı sistemi ile temsil edilmektedir.

TABLO 1
AES-128 anahtar genişletme rutininde kullanılan döngü sabitleri

Döngü	Döngü Sabiti (RCon)	Döngü	Döngü Sabiti (RCon)
1	$(01\ 00\ 00\ 00)_h$	6	$(20\ 00\ 00\ 00)_h$
2	$(02\ 00\ 00\ 00)_h$	7	$(40\ 00\ 00\ 00)_h$
3	$(04\ 00\ 00\ 00)_h$	8	$(80\ 00\ 00\ 00)_h$
4	$(08\ 00\ 00\ 00)_h$	9	$(1B\ 00\ 00\ 00)_h$
5	$(10\ 00\ 00\ 00)_h$	10	$(36\ 00\ 00\ 00)_h$



Şekil 2. AES -128 için anahtar genişletme rutini (N_r döngü sayısını temsil etmektedir)

III. AES ANAHTAR GENİŞLETME RUTİNİNDEKİ İKİ ÖNEMLİ EKSİKLİK

Bir blok şifre daha önce de belirtildiği gibi döngülerden ve döngülerdeki aynı adımlardan oluşmaktadır. Dolayısıyla döngülerdeki simetriyi bozmak için her döngüde farklı bir anahtar materyalinin kullanılması gereklidir. Anahtar genişletme rutinleri gizli anahtardan her döngüde kullanılacak farklı anahtarların (alt anahtarların) elde edilmesini sağlayan algoritmalardır. Her blok şifrede farklı rutinler kullanılabilmektedir ve şifreleme algoritmasında kullanılan yapılar tercih edilerek bu rutinler geliştirilebilir. Lars Knudsen [16] güçlü bir anahtar genişletme rutininin özelliklerini aşağıdaki gibi vermektedir:

1- Çarpışmaya dayanıklı tek yönlü fonksiyon (one-way function) olma,
2- Tüm alt anahtarlar ve gizli anahtar arasında minimum karşılıklı ilişki bulunma,
3- Uygulama etkinliği.

Tüm alt anahtarlar ve gizli anahtar arasında minimum karşılıklı ilişki özelliği blok şifreler üzerine saldırı senaryolarının karmaşıklığını azaltarak saldırgana yardımcı olacak ilişkileri yok edecektir [15]. Bu tür ilişkilerin kullanıldığı saldırılara örnekler DES blok şifresine karşı doğrusal kriptanaliz [17], diferansiyel kriptanaliz [18] gibi saldırılar ile AES blok şifresine karşı olan ilişkili anahtar saldırısı tabanlı çeşitli saldırılar verilebilir. Yine [12] çalışmasının yazarları "Bazı saldırıların genişletilen anahtar bitleri arasındaki ilişkileri kullandıklarını ve bu ilişkilerin olamaması durumunda saldırıların daha yüksek karmaşıklık gerektireceğini" belirtmişlerdir.

Şifreleme algoritması ve anahtar genişletme algoritması güvenlik açısından olduğu kadar uygulama yönüyle de birbirlerini tamamlamalıdırlar. Bu açıdan bakıldığında anahtar genişletme algoritmasında, şifreleme algoritmasında kullanılan optimize edilen elemanların tekrar kullanılması bir avantaj olarak kabul edilebilir [15].

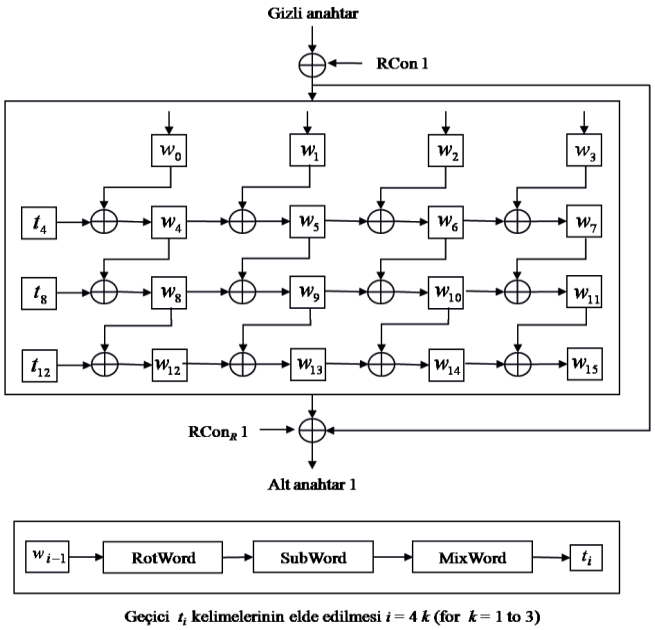
Anahtar genişletme algoritmaları ile elde edilen alt anahtarların üzerinde yürütülen iki önemli test, frekans testi ve çığ kriteri testidir. Frekans testi, bit karıştırma özelliğinin ölçülmesinde (Shannon'nın karıştırma özelliğinin ölçülmesinde temel teşkil eder) kullanılırken çığ kriteri testi, bit yayılım özelliğinin ölçülmesinde kullanılır. Bu test, giriş bloğunda bir bit değişimin çıkış bloğundaki bitlerin yarısının değişimini kontrol eder (Shannon'nın yayılım özelliğinin ölçümünü sağlar).

AES-128 blok şifresinin anahtar genişletme algoritması düşünüldüğünde, Şekil 2'de genel formu verilmiştir, yukarıda verilen özelliklerden sadece üçüncü özelliği sağladığı [15] belirtilmiştir. Bunun yanında AES'in anahtar genişletme algoritmasının kötü yayılım özelliği ilişkili anahtar saldırıları gibi bazı saldırılarda etkin olarak kullanılmaktadır. Bu tür saldırılar gerçek hayatta her ne kadar pratik olmasalar da AES-192 (192-bit anahtar kullanan AES blok şifresi) ve AES-256 (256-bit anahtar kullanan AES blok şifresi) için ilişkili anahtar saldırıların ne kadar faydalı olduğu [19,20] çalışmalarında gösterilmiştir. Bunun temel nedeni olarak AES-192 ve AES-256 versiyonlarındaki anahtar planlama algoritmasının AES-128 (128-bit anahtar kullanan AES blok şifresi) versiyonuna göre daha yavaş yayılım özelliği sağlaması olarak verilebilir. Ayrıca zaman karmaşıklığı açısından Biryukov vd. [11] 10 döngüye kadar bir AES algoritmasına pratik bir saldırıyı göstermişlerdir. Diğer yandan AES'in anahtar genişletme

algoritmasında bit sızıntısı (bit leakage) problemi bulunmaktadır. Bu problem kullanılarak çeşitli saldırılarda bir alt anahtardan faydalanarak diğer alt anahtardan parçalar elde edilebilmektedir. Örneğin [14] çalışmasında bu sızıntı problemi imkânsız diferansiyel saldırısında kullanılmıştır. Bu problemin önüne geçmek için alt anahtarların birbirinden bağımsız olarak üretilmesi bir yöntem olarak kullanılabilir.

IV. AES ANAHTAR GENİŞLETME RUTİNİNİN GÜÇLENDİRİLMESİ

Bölüm 3'te ifade edildiği gibi AES anahtar genişletme rutini iki önemli zaafa sahiptir (yavaş yayılım ve bit sızıntı). Bu zaaflardan rutindeki yavaş yayılımın giderilmesi için öncelikle geçici t_i değerleri elde edilirken bir yayılım elemanına daha ihtiyaç duyulduğu gözlenmektedir. Dolayısıyla bu eksik yayılım elemanı AES şifresinin döngü fonksiyonunda kullanılan MixColumns (Sütunları Karıştırma) dönüşümünün kullanılması ile giderilebilir. Diğer yandan Şekil 3'te verilen yeni anahtar genişletme rutininde gösterildiği gibi 3 defa orijinal AES anahtar genişletme rutininin yürütülmesi ve geçici t_i değerlerinin elde edildiği yeni yapının kullanımı ile iyi yayılım sağlayan bir rutin elde edilebilir. Alt anahtarların elde edilmesi esnasında, gizli anahtar ile XOR'lama işlemine giren döngü sabitinin sonuca eklenmesi bit sızıntı problemini de giderecektir. Yine farklı alt anahtarların elde edilmesi için kullanılan döngü sabitinin ters sırada ($RCON_R$ 1) sonuca eklenmesi ile RCON değerlerinin gizli anahtar olması durumunda bu gizli anahtarların farklı alt anahtarlar üretebilmesi sağlanacak ve oluşabilecek bir simetri giderilebilecektir. RCON 1 değeri $(r_0, r_1, \ldots, r_{15})$ şeklinde 16 byte değerini temsil ederse, $RCON_R$ 1 değeri $(r_{15}, r_{14}, \ldots, r_0)$ 16 byte değerin ters sırasını temsil eder. Sonuç olarak Şekil 3'teki verilen yeni anahtar genişletme rutininde diğer anahtarların elde edilmesi sadece farklı RCON değerlerinin bu yapıya uygulanması ile sağlanacaktır. AES-192 ve AES 256 versiyonları için anahtar genişletme rutini Şekil 3'te verilen yapı ile aynı olmakla beraber döngü sabiti değerleri sırasıyla 192-bit ve 256-bit değerlere sahip olacaktır. Buna ek olarak 3 defa uygulanan orijinal AES anahtar genişletme rutinindeki 4 kelime yerine sırasıyla 6 ve 8 kelime kullanılacaktır.



Şekil 3. AES -128 için önerilen anahtar genişletme rutini

TABLO 2
AES-128 için önerilen anahtar genişletme rutini için ilk alt anahtarın elde edilmesi

Gizli anahtar	01234567 89ABCDEF EDCBA998 76543210
RCon 1	9CC35F04 6A78F579 FC42EC14 2FCDB104
(w_0, w_1, w_2, w_3)	9DE01A63 E3D33896 1189458C 59998314
t_4	D9F3ABB2
(w_4, w_5, w_6, w_7)	4413B1D1 A7C08947 B649CCCB EFD04FDF
t_8	3605A92F
$(w_8, w_9, w_{10}, w_{11})$	721618FE D5D691B9 639F5D72 8C4F12AD
t_{12}	A2CDD003
$(w_{12}, w_{13}, w_{14}, w_{15})$	D0DBC8FD 050D5944 66920436 EADD169B
$RCon_R 1$	04B1CD2F 14EC42FC 79F5786A 045FC39C
Alt anahtar 1	498A1FB1 F232232E 0EEE39D0 B71B5613

Örnek 1. Tablo 2'de verilen 128-bit büyüklüğündeki bir gizli anahtardan blok şifre için ilk alt anahtarın elde edilmesi gösterilmektedir.

Örnek 2. Tablo 3'te verilen aralarında 1-bit değişime sahip iki gizli anahtar için alt anahtarlar arasındaki bit değişimi (B.D.) gösterilmektedir.

Örnek 3. Tablo 4'te AES-128 için önerilen anahtar genişletme rutini için her gizli anahtarın 128 farklı bit pozisyonunda 1-bit değişimi sonucu elde edilen ortalama bit değişimi verilmiştir. Ayrıca bu ortalamalar her döngü anahtarı ve farklı 20 gizli anahtar için elde edilmiştir. Sonuçlar çığ özelliği açısından iyi sonuçlar vermektedir.

TABLO 3

AES-128 için önerilen anahtar genişletme rutini ile üretilmiş bir bit değişime sahip iki gizli anahtardan elde edilen döngü anahtarları kümesinin bit değişimleri açısından karşılaştırılması

R	Döngü anahtarları kümesi 1				Döngü anahtarları kümesi 2				B.D.
-	01234567	89ABCDEF	EDCBA998	76543210	41234567	89ABCDEF	EDCBA998	76543210	1
1	498A1FB1	F232232E	0EEE39D0	B71B5613	8A4F5F2A	D75192E2	CD2B794B	D278E7DF	59
2	E6B8305E	46FAAE29	2D3E7F6A	6C96E002	EE1DE3E3	5203992C	259BACD7	386FD707	63
3	7AACD838	04288D28	28DA4722	46724914	11FCDB37	0453F371	438A442D	0609374D	59
4	0C597D7A	355C1604	29349D75	713AA8FB	7B429B6F	2C69824D	5E2F7B60	280F3CB2	63
5	A5B328CB	8604375D	B4544DCD	2CA5A72D	44F7251E	A1C6AB9F	55104018	4B673BEF	57
6	CDEB5C31	5DA9E411	CD41C2F6	AF9384F2	9A07B592	CCC39CA5	9AAD2B55	7EF9FC46	69
7	B9BCCB90	2473B523	B4A11999	31951DF4	439CCE15	B27FC4FE	4E811C1C	E7996C29	57
8	7B22245A	1AFECC32	7E4F6985	B544B21D	4E9BAB10	87AF60B3	4BF6E6CF	68151E9C	63
9	6C6AF1E8	31EA58D9	4916923E	7ECECE41	EFF2F107	76F6CF25	CA8E92D1	79D259BD	61
10	2FB5795C	9E4456B8	44878B28	E868E2F2	43967E85	568366B2	28A48CF1	60AFD2F8	53

TABLO 4

AES-128 için önerilen anahtar genişletme rutini ile 20 gizli anahtardan üretilmiş alt anahtarların 128 farklı bit pozisyonu için ortalama bit değişimleri

Gizli Anahtar	Döngü Anah.1	Döngü Anah. 2	Döngü Anah. 3	Döngü Anah. 4	Döngü Anah. 5	Döngü Anah. 6	Döngü Anah. 7	Döngü Anah. 8	Döngü Anah. 9	Döngü Anah. 10
Anah. 1	63.44	64.58	63.84	62.28	62.50	63.77	63.55	64.50	63.98	63.55
Anah. 2	64.91	64.53	65.48	63.19	63.23	63.73	64.45	64.23	64.63	64.08
Anah. 3	64.02	63.52	63.75	63.58	63.69	65.25	64.13	63.33	65.06	63.64
Anah. 4	64.02	64.25	64.20	63.53	63.09	64.75	64.06	64.81	64.72	63.89
Anah. 5	64.30	63.80	62.34	62.66	62.14	64.97	63.59	63.03	62.41	63.33
Anah. 6	64.75	63.38	64.64	63.13	64.58	64.55	64.39	64.22	64.75	64.16
Anah. 7	64.53	63.88	64.52	64.17	63.61	63.66	65.33	64.59	63.69	62.53
Anah. 8	65.00	64.13	63.42	63.38	64.06	64.00	63.44	63.03	63.55	62.69
Anah. 9	62.81	64.08	62.52	64.13	64.84	64.36	64.02	63.22	63.38	62.05
Anah.10	63.58	63.75	64.67	64.59	63.92	63.97	64.13	64.34	65.25	63.61
Anah. 11	63.91	63.97	63.88	62.00	64.36	63.91	64.36	62.13	63.13	62.45
Anah. 12	65.02	64.05	65.06	64.09	64.70	62.78	63.36	64.19	64.19	65.13
Anah. 13	62.91	65.70	62.83	64.33	64.33	62.39	63.66	63.61	63.25	62.28
Anah. 14	63.72	62.23	63.55	62.22	64.91	64.27	64.89	65.08	62.61	63.45
Anah. 15	63.73	64.59	62.72	63.91	65.89	64.50	65.11	63.38	64.72	63.61
Anah. 16	63.52	63.88	63.03	64.16	65.22	63.16	65.67	63.14	65.34	63.48
Anah. 17	63.42	64.52	64.50	64.19	62.63	64.38	64.38	63.22	63.73	62.59
Anah. 18	64.59	63.63	63.13	62.19	64.02	63.47	64.23	63.94	62.47	65.63
Anah.19	63.02	64.48	64.41	62.78	64.39	64.58	65.06	64.50	64.84	63.38
Anah. 20	65.89	63.89	63.13	63.69	65.09	63.92	63.81	64.28	63.19	63.42
Ortalama	64.05	64.04	63.78	63.41	64.06	64.02	64.28	63.79	63.94	63.45
% Ortalama	50.04	50.03	49.83	49.54	50.05	50.01	50.22	49.83	49.96	49.57

V. BİR BLOK ŞİFREDEN BAĞIMSIZ ANAHTAR GENİŞLETME RUTİNİN TASARIMI

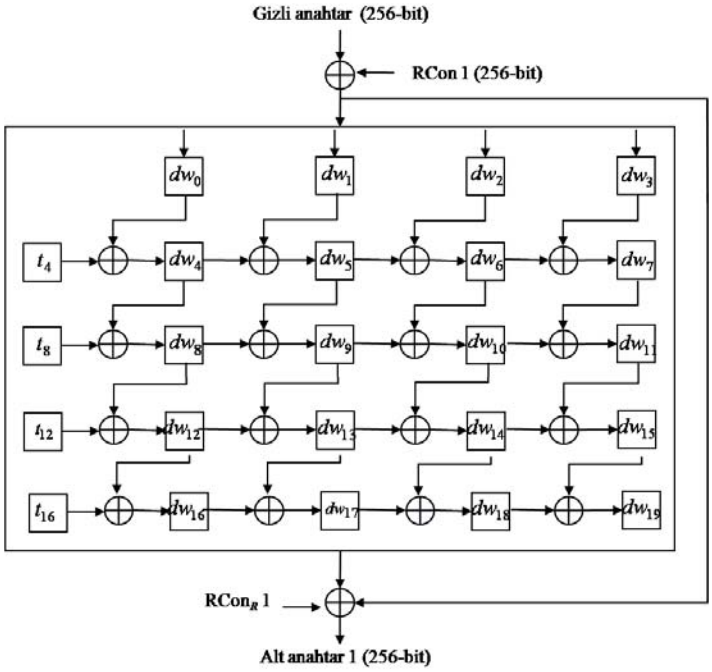
Bölüm 4'te AES blok şifresi için öne sürülen rutinden yola çıkarak bir blok şifreden bağımsız anahtar genişletme rutini mimarisi ortaya konabilir. Günümüzde kullanılan blok şifreler genellikle kendi bünyelerinde bulunan elemanları kullanan anahtar genişletme rutinlerine sahiptir. Bununla beraber bu blok şifrelerin döngü fonksiyonlarının tasarımında yer değiştirme katmanı olarak kullanılan S-kutuları 4-bit (lightweight-hafif siklet blok şifreler) ya da 8-bit boyutunda olacak şekilde tercih edilmektedir. Diğer yandan uygulama etkinliği anahtar genişletme rutinleri için önemli bir kriterdir. Şekil 3'te verilen mimaride geçici t_i değerlerinin elde edilmesi için iki farklı mimari ortaya konabilir:

1- Yayılım –Yer değiştirme – Yayılım (DSD: Diffusion-Substitution-Diffusion),
2- Yer değiştirme –Yayılım – Yer değiştirme (SDS: Substitution-Diffusion-Substitution).

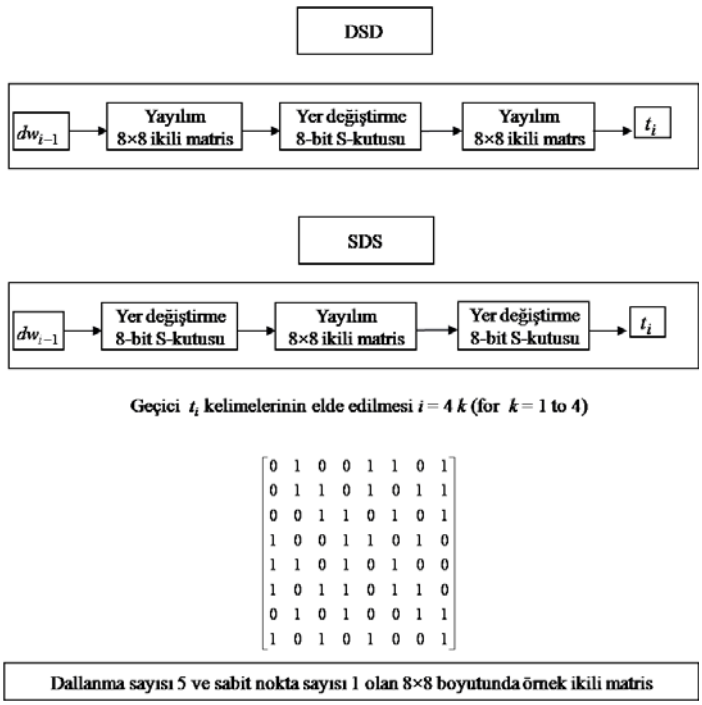
Verilen bu iki mimari de yayılım katmanının iyi uygulama özelliğine, yüksek dallanma sayısı (branch number) değerine ve olabildiğince az sabit nokta sayısına [21] sahip olması (1 sabit nokta) istenebilecek özellikler olarak karşımıza çıkacaktır. Blok şifreden bağımsız tasarlanacak anahtar genişletme rutininden farklı boyutlarda alt anahtarlar üretilebilmesi ve genişletme rutininin S-kutusunun büyüklüğünden bağımsız olabilmesi iyi bir yayılım elemanın seçimi ile sağlanabilir. Dolayısıyla bahsedilen özellikleri karşılayabilecek yayılım elemanı olarak ikili matrisler (dallanma sayısı yüksek, sabit nokta sayısı düşük ve sadece XOR işlemi tabanlı) seçilebilir. Örneğin 4×4, 5×5, 8×8 boyutunda ikili matrislerin maksimum dallanma sayıları sırasıyla 4, 4 ve 5'tir [22] ve sadece XOR işlemi tabanlı olarak uygulamaları gerçekleştirilebilir. Sahip olunan S-kutusu büyüklüğü ve üretilecek alt anahtar büyüklüğüne göre Şekil 3'te verilen yapı, geçici t_i değerlerinin elde edilmesi için iki farklı mimariden birinin seçilmesi ve uygun özelliklerde ikili matrisin kullanılması ile istenen boyutta alt anahtar üreten bir anahtar genişletme rutini haline getirilebilir.

Örnek 4. 256-bit alt anahtar üreten blok şifreden bağımsız bir anahtar genişletme rutinin tasarımı için elimizde 8-bit S-kutusu (AES S-kutusu), 8×8 boyutunda (dallanma sayısı

değeri 5 ve 1 sabit noktaya sahip [23]) ikili matris olsun. Buna ek olarak Şekil 3'te AES-128 için önerilen rutinde 32-bit geçici t_i değerleri yerine 64-bit t_i değerleri kullanılsın. Orijinal AES anahtar rutininin yürütülme sayısı 4 olarak seçilsin (Tekrar etme işleminin sayısı deneysel sonuçlarla belirlenmesi daha uygun olacaktır). Şekil 4 ve Şekil 5'te bir blok şifreden bağımsız anahtar genişletme rutini, bu rutinde kullanılacak geçici t_i değerlerinin elde edilmesinde kullanılacak iki mimari ve örnek bir 8×8 boyutunda (dallanma sayısı değeri 5 ve 1 sabit noktaya sahip) ikili matris verilmektedir.



Şekil 4. Blok şifreden bağımsız 256-bit anahtar genişletme rutini (dw: double word-64-bit değeri temsil eder)



Şekil 5. 256-bit anahtar genişletme rutininde geçici t_i değerlerinin elde edilmesinde kullanılabilecek iki mimari

VI. SONUÇLAR

Bu çalışmada AES blok şifresinde kullanılan anahtar genişletme rutini incelemiş ve bu rutindeki problemler irdelenmiştir. AES blok şifresinin anahtar genişletme rutinindeki zaafları gideren ve orijinal yapı üzerinde basit değişiklikler yapılarak elde edilen gelişmiş yeni bir anahtar genişletme rutini ortaya konmuştur. Çalışmanın son bölümünde de bu geliştirilen rutinden faydalanılarak bir blok şifreden bağımsız çeşitli büyüklüklerde alt anahtar üreten anahtar genişletme rutininin nasıl tasarımının yapılacağı tartışılarak bu tip bir anahtar genişletme rutini için örnek verilmiştir.

KAYNAKLAR

[1] US National Institute of Standards and Technology, Data Encryption Standard, Federal Information Processing Standards Publications, No. 46-3, 1999.

[2] US National Institute of Standards and Technology, Advanced Encryption Standard, Federal Information Processing Standards Publications}, No. 197, 2001.

[3] B. A. Forouzan, Cryptography and Network Security, McGraw-Hill International Edition, 2008.

[4] B. Aslan, M. T. Sakallı, E. Buluş, Üs Haritalama Tabanlı Cebirsel 8-bit giriş 8-bit çıkışlı S-kutularının Sınıflandırılması, Ağ ve Bilgi Ulusal Sempozyumu 2, Girne-Kıbrıs, 2008.

[5] B. Aslan, M. T. Sakallı, E. Buluş, Classifying 8-bit to 8-bit S-boxes based on Power Mappings from the point of DDT and LAT Distributions, In Proceedings of International Workshop on the Arithmetic of Finite Fields, WAIFI 2008, Lecture Notes in Computer Science, Vol. 5130, Springer-Verlag, 2008; 123-133.

[6] G. Jakimoski, Y. Desmedt, Related-Key Differential Cryptanalysis of 192-bit Key AES Variants, In Proceedings of Selected Areas in Cryptography (SAC 2003), Lecture Notes in Computer Science, Vol. 3006, Springer-Verlag, 2004; 208-221.

[7] E. Biham, O. Dunkelman, N. Keller, Relate-Key Impossible Differential Attacks on 8-Round AES-192, In Proceedings of Topics in Cryptology-CT-RSA 2006, The Cryptographers' Track at the RSA Conference 2006, Lecture Notes in Computer Science, Vol. 3860, Springer-Verlag, 2006; 21-33.

[8] W. Zhang, W. Wu, L. Zhang, D. Feng , Improved Related-Key Impossible Differential Attacks on Reduced-Round AES-192, In Proceedings of Selected Areas in Cryptography (SAC 2006), Lecture Notes in Computer Science, Vol. 4356, Springer-Verlag, 2007; 15-27.

[9] E. Biham, O. Dunkelman, N. Keller, Related-Key Boomerang and Rectangle Attacks., In Proceedings of EUROCRYPT 2005, Lecture Notes in Computer Science, Vol. 3494, Springer-Verlag, 2005; 507-525.

[10] J. Kim, S. Hong, B. Preneel, Related-Key Rectangle Attacks on Reduced AES-192 and AES-256, In Proceedings of FSE 2007, Lecture Notes in Computer Science, Vol. 4593, Springer-Verlag, 2007; 225-241.

[11] A. Biryukov, O. Dunkelman, N. Keller, D. Khovratovich, A. Shamir, Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds, Cryptology ePrint Archive, Report 2009/374, 2009. Available at http://eprint.iacr.org/2009/374/.

[12] N. Ferguson, J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, D. Whiting, Improved Cryptanalysis of Rijndael, In Proceedings of FSE 2000, Lecture Notes in Computer Science, Vol. 1978, Springer-Verlag, 2001; 213-230.

[13] E. Fleischmann, M. Gorski, S. Lucks, Attacking 9 and 10 Rounds of AES-256, In Proceedings of ACISP 2009, Lecture Notes in Computer Science, Vol. 5594, Springer-Verlag, 2009; 60-72.

[14] RC-W. Phan, Impossible differential cryptanalysis of 7-round Advanced Encryption Standard (AES), Information Processing Letters, 2004; 91 (1):33-38.

[15] L. May, M. Henricksen, W. Millan, G. Carter, E. Dawson, Strengthening the Key Schedule of the AES, In Proceedings of ACISP 2002, Lecture Notes in Computer Science, Vol. 2384, Springer-Verlag, 2002; 226-240.

[16] L. Knudsen, Practically Secure Feistel Ciphers, In Proceedings of FSE 1993, Lecture Notes in Computer Science, Vol. 809, Springer-Verlag, 1993; 211-221.

[17] M. Matsui, Linear Cryptanalysis Method for DES Cipher, In Proceedings of EUROCRYPT 93, Lecture Notes in Computer Science, Vol. 765, Springer-Verlag, 1994; 386-397.

[18] E. Biham, A. Shamir, Differential Cryptanalysis of DES-like Cryptosystems, In Proceedings of CRYPTO'90, Lecture Notes in Computer Science, Vol. 537, Springer-Verlag, 1990; 2-21.

[19] A. Biryukov, D. Khovratovich, Related-key Cryptanalysis of the Full AES-192 and AES-256. Cryptology ePrint Archive, Report 2009/317, 2009. Available at http://eprint.iacr.org/2009/317/.

[20] A. Biryukov, Khovratovich D. Related-key Cryptanalysis of the Full AES-192 and AES-256. In Proceedings of ASIACRYPT 2009, Lecture Notes in Computer Science, Vol. 5912, Springer-Verlag, 2009; 1-18.

[21] M.R. Z'aba, Analysis of Linear Relationships in Block Ciphers, Ph.D. Thesis, Queensland University of Technology, Brisbane, Australia, 2010.

[22] D. Kwon, S. H. Sung, J. H. Song, S. Park, Design of Block Ciphers and Coding Theory, Trends in Mathematics, 2005; 8(1):13-20.

[23] B. Aslan, M. T. Sakallı, Algebraic construction of cryptographically good binary linear transformations, Security and Communication Networks (2012) doi:10.1002/sec.556.