YAN-KANAL ANALİZİ SALDIRILARINA GENEL BAKIŞ

LEVENT ORDU , SIDDIKA BERNA ÖRS YALÇIN
lvordu@uekae.tubitak.gov.tr, Siddika.Ors@itu.edu.tr

ÖZET : Kriptografik algoritmaları gerçekleyen donanımlar, açık veya kapalı metin dışında, bazı istemsiz çıkışlar da üretmektedir. Bu istemsiz çıkışlar; işlem süresi, dinamik güç tüketimi, elektromanyetik radyasyon ve cihazın çıkardığı ses olabilir. Eğer böyle bir çıkış, cihaz içinde saklanan gizli bilginin tamamıyla veya bir parçasıyla ilişkiliyse, yan-kanal bilgisi olarak adlandırılır. Yan-Kanal Analizi Saldırıları'nda, bu yan-kanal bilgileri kullanılarak gizli bilgiye ulaşmaya çalışılır. Yan kanal analizi saldırıları, kriptografik algoritmaların gerçeklendiği sistemler için büyük bir tehdit oluşturmaktadır. Bu konu üzerine artarak devam eden araştırmalarda, DES, AES ve RSA'nın da içlerinde olduğu bir çok algoritma gerçeklemesinin yan kanal analizi saldırılarına açık olduğu gösterilmiş ve alınabilecek çeşitli önlemler ileri sürülmüştür.

ANAHTAR KELİMELER: Kriptanaliz, Yan-Kanal Analizi Saldırıları, DPA

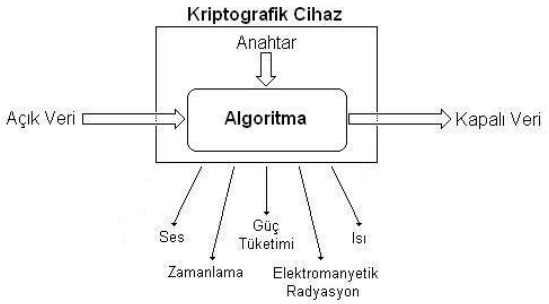
A SURVEY of SIDE-CHANNEL ANALYSIS ATTACKS

ABSTRACT : Implementations of cryptographic algorithms have some unintentional outputs which are neither plaintext nor ciphertext. These side outputs (timing information, power consumption, electromagnetic radiation and acustic information) are called side-channel informations if they are somehow related with secret data. Side-Channel Analysis (SCA) Attacks use these side-channel informations to recover secret information. Side-channel attacks pose a serious threat to the implementations of cryptographic algorithms. It has been shown that, many implementations of different algorithms are susceptible to side-channel attacks, including DES, AES and RSA. This is why there has been lots of research to develop countermeasures against theese attacks.

KEYWORDS : Cryptanalysis, Side-Channel Analysis Attacks, DPA

1 Giriş

Kriptografik algoritmaların gerçeklemeleri olan cihazlar, açık veri ve kapalı veri dışında, istemsiz bazı çıkışlar (Şekil 1) da üretmekte ve bu bilgiler kolaylıkla ölçülebilmektedir. Örneğin bir işlemin yapılmasının ne kadar zaman aldığı, cihazın ne kadar dinamik güç harcadığı, ne kadar elektromanyetik yayınım yaptığı, nasıl ve ne şiddette sesler çıkardığı, veya ne kadar ısı yaydığı bunlardan en bilinenleridir.



Şekil 1

Eğer bu çıkışlar cihazın içinde saklanan gizli bilgilerle bir şekilde bağlantılılarsa yan-kanal bilgisi olarak adlandırılırlar. Yan-kanal analizi saldırıları, kriptografik cihazın ürettiği yan-kanal bilgilerini kullanarak gizli bilgiye ulaşmaya çalışır. Aynı algoritmanın farklı gerçeklemeleri değişik miktar ve biçimlerde yan kanal bilgisi sızdırabilir. Bu nedenle, çoğunlukla, yan-kanal analizi saldırıları genelleştirilemezler. Buna karşın bu saldırılar genellikle pratikte kullanılmaya uygundur.

Yan-kanal analizi saldırıları, aktif ve pasif olarak iki gruba ayrılmaktadır. Aktif saldırılar ya da diğer adıyla kurcalama saldırıları [1], kriptografik cihazın içindeki devrelere ulaşılmasını gerektirir. Bu nedenle uygulanmaları daha zordur ve oldukça gelişmiş ve pahalı düzeneğe ihtiyaç duyulur. İki tür aktif saldırı vardır; ölçüm saldırıları [2] ve hata-oluşturma saldırıları [3, 15]. Ölçüm saldırılarında, saldırgan, cihaz içindeki devrelere erişip bellek bölgelerini okuyarak yada veri iletim hatlarını gözleyerek doğrudan gizli bilgiye erişmeye çalışır. Hata

oluşturma saldırılarında ise belirli noktalara dışarıdan müdahale edilip, işlemlerde hataya yol açarak gizli bilgiler elde edilmeye çalışılır.

Pasif saldırılar 1996'da zamanlama analizi ile ilgili ilk makale yayınlandığında ilk defa önemli bir tehdit olarak görülmeye başlandı [4]. Pasif saldırılarda cihazın çalışmasına müdahale edilmez. Cihazın normal çalışması sırasında ürettiği yan-kanal bilgileri kullanılır. Bu saldırılar çok daha basit ölçüm düzenekleriyle yapılabilmektedir. Pasif saldırılar kullandıkları yan-kanal bilgisine göre dört gruba ayrılır; Zamanlama Analizi Saldırıları, Güç Analizi Saldırıları, Elektromanyetik Analiz Saldırıları ve Akustik Analiz Saldırıları.

Bu bildiride Yan-Kanal Analizlerinin en bilinenleri olan; Zamanlama Analizi, Güç Analizi ve Elektromanyetik Analiz sırasıyla 2, 3 ve 4 numaralı bölümlerde genel olarak tanıtılmış ve mümkün olduğunca uygulamalarıyla açıklanmaya çalışılmıştır.

2 Zamanlama Analizi Saldırıları

Zamanlama analizi (ZA) saldırılarında, sabit veri işleme zamanına sahip olmayan algoritmaların sızdırdığı zamanlama yan-kanal bilgisinden faydalanılır. Yan-kanal bilgisi oluşmasının nedeni, algoritmanın adımlarının birinde yürütülen bir işlemin süresinin kullanılan gizli anahtara bağımlı olmasıdır [4, 16]. Bu, anahtara bağlı olarak yapılan dallanma işlemlerinden, farklı karmaşıklıklarda işlemler kullanılmasından, işlemlerin gerçeklenmesinde kullanılan eniyileştirme (optimization) tekniklerinden veya önbellek kullanımından kaynaklanabilir. Özellikle asimetrik anahtarlı algoritmalar için bu durum geçerlidir. Simetrik anahtarlı algoritmaların zamanlama karakteristikleri, asimetrik-anahtarlı algoritmalar kadar anahtara bağımlı olmadığı için zamanlama analizi saldırılarına karşı daha güçlüdürler.

Örneğin toplama ve çarpma işlemlerinin yürütülmesi genellikle farklı zamanlarda tamamlanabilmektedir. x, y m-bitlik değişkenler olmak üzere, $z = x + y$ ve $z = x \times y$ işlemlerinin hesaplandığını düşünelim. Toplama işlemi $T_T = m$ saat darbesi sürede tamamlansın. Eğer çarpma işlemi, toplama işlemini taban alarak hesaplanıyorsa, bu durumda çarpma işlemi,

$$T_Ç = \frac{3 \times (m-1) \times m}{2}$$

sürede hesaplanır. Görüldüğü gibi, aynı bit-sayısına sahip iki terim kullanan işlemler farklı sürelerde tamamlanmaktadır. Bundan faydalanarak, saldırgan, yürütülen işlemin süresine bakarak toplama mı yoksa çarpma mı yapıldığını anlayabilir [14].

İşlemlerin yürütülme sürelerindeki farklardan faydalanarak elde edilen zamanlama yan-kanal bilgisi Kocher ve Janke ve Lehmann'ın çalışmalarında gizli bilgiye ulaşmak amacıyla başarıyla kullanılmıştır [4,5]. Özellikle Kocher'in çalışması yan-kanal analizi araştırmaları için önemli bir temel oluşturmaktadır. Bölüm 2.1'de kısaca bu çalışmaya değinilmiştir.

2.1 Modülo Üs Alıcılara ZA Saldırısı

RSA algoritmasında gizli-anahtarın kullanıldığı işlem,

$R = f(x, y) = y^x \mod n$,

şeklindedir. Burada; n bilinen bir değerdir, y ise giriş değeridir. Saldırganın amacı gizli anahtar bilgisi olan x'in bulunmasıdır.

[4]'te önerilen saldırıda kripto cihazına, $y^x \bmod n$ değeri, k adet farklı y değeri için hesaplattırılır. Tüm y değerlerine karşı düşen işlem süreleri saldırgan tarafından kaydedilir. İşlemlerin süresi, saldırgan tarafından, girişlerin hedeflenen cihaza ulaşmasıyla çıkışın üretilmesi arasında geçen zamana bakarak ölçülebilir. Ayrıca saldırının başarılı olabilmesi için, tüm işlemler boyunca aynı x değerinin kullanılması gereklidir.

Saldırı, sabit süreli olmayan işlemler içeren herhangi bir yapı için çalışabilmektedir. Örneğin aşağıdaki, $R = y^x \mod n$ (x, w-bit uzunluğunda olmak üzere), değerini hesaplamakta kullanılan algoritma ele alınırsa;

$s_0 = 1$,
$0 \le k \le (w-1)$ *için*;
  Eger $x_k = 1$ *ise*,
    $R_k = (s_k \times y) \bmod n$
  Degilse,
    $R_k = s_k$
  $s_{k+1} = R_k^{\,2} \mod n$
Sonuc $= R_{w-1}$

Saldırı yöntemi kullanılarak, üssün ilk b biti biliniyorsa, $(b+1).$ biti elde edilebilir. Bu şekilde tüm kuvvet terimi bitleri elde edilir. İlk b bit değeri bilindiği için, algoritmanın ilk b adımı hesaplanarak s_b değeri bulunur. Sonraki adım, ilk bilinmeyen bit değerini gerektirir. Eğer bu bit '1' ise, $R_b = (s_b \times y) \bmod n$ işlemi yapılır, bit '0' ise bu işlem atlanır. Saldırıda bu dallanmadan faydalanılır.

Bazı (s_b, y) değerleri için, $R_b = (s_b \times y) \bmod n$ işlemi normalde aldığından çok daha uzun süre alır. Saldırgan sistemin yapısını inceleyerek bu değerleri hesaplayabilir. Eğer, $R_b = (s_b \times y) \bmod n$ hesabını

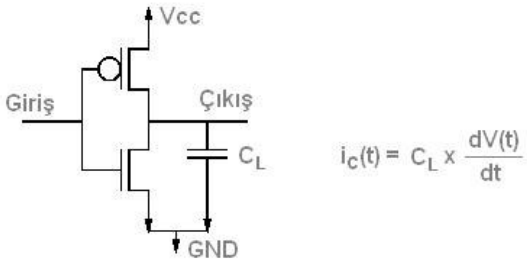
çok yavaşlatan bir y değeri için, toplam üs alma işlemi kısa sürede tamamlanıyorsa, b indisli anahtar biti '0' olmalıdır. Çünkü $R_b = (s_b \times y) \bmod n$ hesabı atlanmış, böylelikle onun yaratacağı fazladan gecikme oluşmamış olacaktır. Benzer şekilde, $R_b = (s_b \times y) \bmod n$ hesabını çok yavaşlatan bir y değeri için, toplam kuvvet alma işlemi normalden daha uzun sürede tamamlanıyorsa, b indisli anahtar biti '1' olmalıdır. Çünkü $R_b = (s_b \times y) \bmod n$ hesabı yapılmış, böylelikle onun yaratacağı fazladan gecikme toplam işlem süresine eklenmiş olacaktır. Anahtarın ilk biti '0' olarak kabul edilerek yukarıdaki şekilde anahtarın tüm bitleri bulunur. Aynı şekilde ilk bit '1' olarak kabul edilerek yukarıdaki şekilde anahtarın tüm bitleri bulunur. İki başlangıç değerinden birisi için doğru sonuca ulaşılır.

2.2 ZA Saldırılarına Karşı Tedbirler

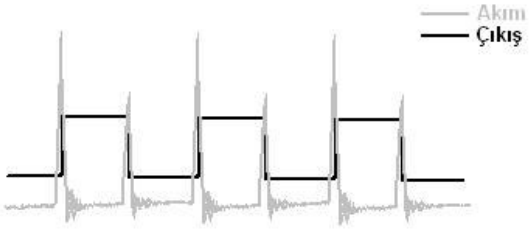
Zamanlama saldırılarına karşı koyabilmek için kriptografik algoritmaların gerçeklemeleri sabit işletim süresine sahip olacak şekilde tasarlanmalıdır. Günümüzde algoritma gerçeklemelerinin hemen hepsi, zamanlama saldırılarına karşı dirençlidir. Zamanlama analizine karşı koyma yöntemlerinden bazıları [17, 18, 19, 20] de bulunabilir. Bununla birlikte, zamanlama bilgisi diğer yan-kanal bilgileriyle birlikte kullanılabilir. Örneğin zamanlama bilgisi, bir algoritmanın özel parçalarını belirlemek amacıyla kullanılabilir.

3 Güç Analizi Saldırıları

Günümüzde tamamlayıcı metal oksitli yarı-iletken (CMOS: complementary metal oxide semiconductor) tranzistörler, elektronik tümdevre gerçeklemelerinde çok yaygın olarak kullanılmaktadır. Bir CMOS tranzistörün güç tüketiminin büyük kısmı dinamik güç tüketimine (konum değiştirme anlarındaki güç tüketimi) bağlıdır. Çünkü, tranzistörün sürdüğü yük kapasitesinin (Şekil 2'de C_L) akımı, üzerindeki gerilimin değişimine bağlıdır [21]. Bu sebeple çıkışın sabit kaldığı anlardaki güç tüketimi, Şekil 3'te gösterildiği gibi, çok düşük kalmaktadır. Yine Şekil 3'ten görüleceği üzere, $0 \rightarrow 1$ geçişlerindeki güç tüketimi, $1 \rightarrow 0$ geçişlerine oranla daha yüksektir.



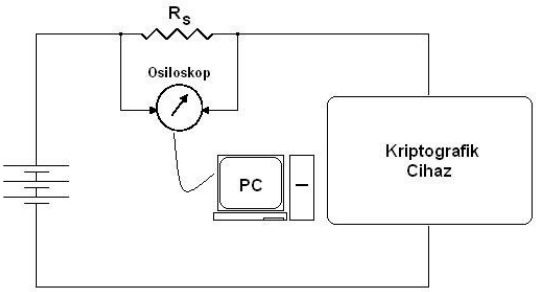
Şekil 2



Şekil 3

Bir lojik kapının güç tüketimi giriş değerleri ile doğrudan ilişkilendirilebilir. Giriş değerinin değişmesi, çıkışın konum değiştirmesine ve dolayısıyla dinamik güç tüketimine neden olabilir. Bu nedenle, CMOS kapıların güç tüketimi, kapı girişleri gizli bir bilgiye bağlıysa, yan-kanal bilgisi olarak kullanılabilir [6]. Farklı işlemlerin farklı güç tüketim karakteristiğine sahip olması, saldırıların başarısını arttırır.

Güç analizi (PA: Power Analysis) saldırılarında [14, 22], kriptografik cihazın güç tüketimi ile gizli bilgi ya da yapılan işlemler arasında bir korelasyon kurularak gizli bilgiye erişilmeye çalışılır. Bunun için öncelikle güç tüketiminin ölçülmesi gerekmektedir. Bu amaçla, devre ile kaynak arasındaki hat üzerine küçük değerli bir direnç yerleştirilir ve bu direncin her iki ucundaki gerilim değerlerinin farkından yararlanılarak çekilen akım bilgisi elde edilir [23]. Şekil 4'ten de görüldüğü gibi, ölçüm düzeneği oldukça basit yapıdadır ve kullanılan düzenek pahalı değildir. Bu da, Güç Analizi (GA) yöntemini aktif saldırılardan daha büyük bir tehdit haline getirmektedir.



Şekil 4

Güç analizi saldırıları ilk kez Kocher [6] tarafından DES üzerinde uygulanmıştır. Bu uygulamanın başarısının ardından, güç analizi üzerine pek çok çalışma yapılmıştır. Güç analizi saldırıları iki başlık altında toplanabilir; basit güç analizi saldırıları ve diferansiyel güç analizi saldırıları.

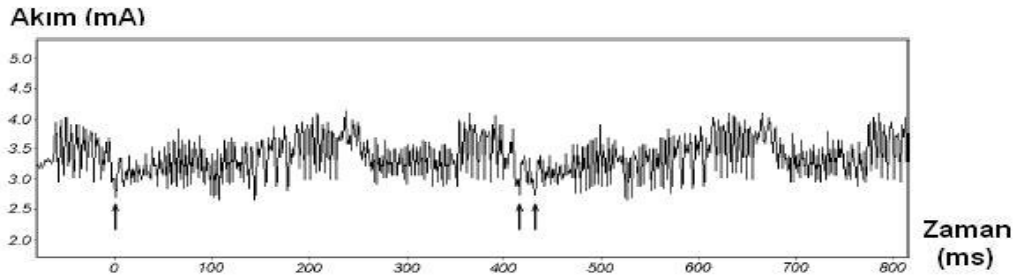
3.1 Basit Güç Analizi Saldırıları

Basit Güç Analizi (BGA) Saldırıları, kripto cihazı çalışırken yapılan tek bir güç tüketimi ölçümü

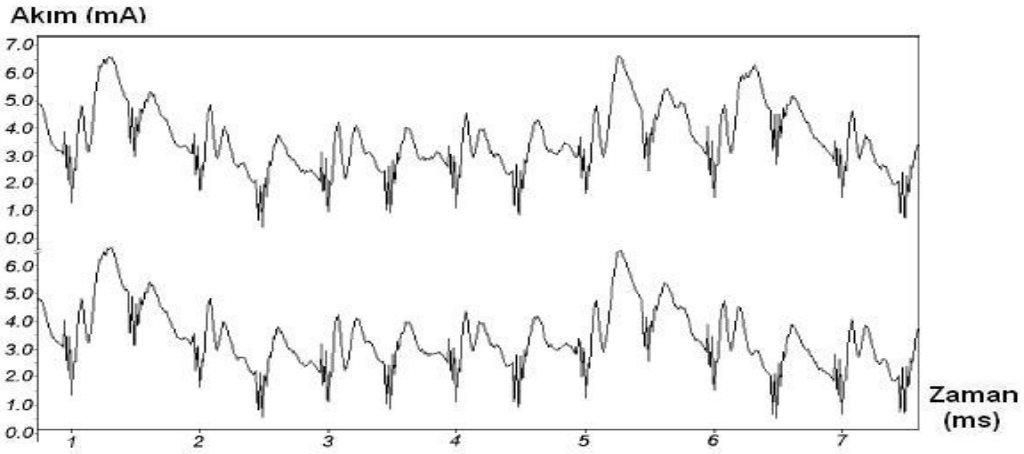
kullanılarak yapılır. Elde edilen ölçümler genellikle gözle incelenerek yorumlanır. BGA'da, yürütülen işlemlerle güç tüketimi yan-kanal bilgisi arasında ilişki kurulmaya çalışılır. Elde edilen ölçümler incelenerek, kripto cihazının işleyişi hakkında bilgi edinileceği gibi, doğrudan gizli anahtar bilgisine de ulaşılabilir.

Farklı mikroişlemci komutları işlenirken, ya da toplama ve çarpma gibi farklı işlemler gerçekleştirilirken entegre devreler farklı miktarda güç tüketirler. Bir koddaki dallanmalar bu tür farklılıkların genel olarak oluştuğu bölümlerdir. Güç tüketimi ölçümleri incelenirken bu faklılıklar kolayca gözlemlenebilir. Hatta RSA'de dallanma sonucunda gerçekleştirilen çarpma ve kare alma işlemleri, yada DES turundaki dallanma sonucunda gerçekleştirilen kaydırma işlemlerinden yararlanılarak bu iki algoritmada kullanılan anahtarlar elde edilebilir [6].

Şekil 5'te gösterilen ardarda gelen iki DES turuna (2. ve 3. turlar) ait yüksek çözünürlüklü bir güç tüketimi ölçümü incelendiğinde, algoritmaya ait bazı ayrıntılar daha açık olarak görülebilmektedir [6]. 28-bitlik DES anahtar bellekleri, 2. turda bir kez (soldaki ok), 3. turda ise iki kez (sağdaki iki ok) kaydırılmaktadır. Gözlemlenebilen bu küçük farklılıklar, çoğunlukla koşullu dallanma işlemlerinden kaynaklanmaktadır.



Şekil 5



Şekil 6

Daha yüksek çözünürlüklü ölçümler incelendiğinde çok daha ayrıntılı bilgilere ulaşılabilir. Şekil 6'da, 7 saat darbelik bölümde yürütülen mikroişlemci kodları için elde edilen iki güç ölçümü gösterilmiştir. Üstteki grafikte, 6. saat darbesinde bir dallanma yapılırken, alttaki grafikte bu dallanma atlanmaktadır. Güç tüketimindeki fark açıkça görülmektedir.

Yürütülen işlemlerin, işlenen veriye bağlı olarak, değiştiği bölümlerin bulunduğu algoritmaların BGA ile kırılması mümkün olabilir [6]. Örneğin; DES anahtar üreteci: DES tur anahtarlarının üretimi sırasında 28-bitlik belleklerde bulunan veriler kaydırılmakta, bu işlem sonucunda en anlamsız bit dışarı verilmektedir. Sonrasında bu bit değerine bağlı olarak işlemler yapılmaktadır. '0' ve '1' bitleri için yapılan işlemlerin güç tüketimlerinin farklı oluşundan faydalanarak anahtar elde edilebilmektedir.

BGA saldırılarından korunmak için algoritma tasarımı sırasında ya da gerçekleme aşamasında bazı önlemler alınabilir [24, 25]. Örneğin; Gizli ara değerler ya da anahtar bilgisi kullanılarak yapılan koşullu dallanmalardan sakınılması, bir çok BGA saldırısının yapılmasını olanaksız kılmaktadır. Dallanmaların şart olduğu durumlarda, algoritmanın gerçeklenmesi aşamasında, güç tüketimini dengeleme yöntemleri kullanılabilir. Bu amaçla, güç tüketiminin düşük

olduğu dallarda, algoritmanın işleyişine etkisi olmayan gereksiz işlemler de yapılarak, bu dalların güç tüketiminin yüksek olduğu dallardan ayırt edilmesi engellenebilir. Mikroişlemcilerde zayıf dallara işlevsiz kodlar (NOP: No Operation) yerleştirilebilir.

3.2 Diferansiyel Güç Analizi Saldırıları

Yürütülen komutlara veya yapılan işlemlere bağlı olarak oluşan güç tüketimi değişimlerine ek olarak, işlenen veriye bağlı olarak da güç tüketiminde değişimler oluşmaktadır. Bu değişimlerin çok düşük boyutta olmaları, ölçüm hataları yada gürültü nedeniyle gözlemlenmelerini zorlaştırır. Ancak yine de, hedeflenen algoritmaya yönelik bazı istatistiksel teknikler ve hata düzeltme yöntemleri kullanılarak yan-kanal bilgisi elde edilebilir ve gizli bilgiye ulaşılabilir.

Diferansiyel Analiz saldırılarında, saldırgan saldırılan cihazın hipotetik bir modelini kullanır. Bu modelin kalitesi saldırganın sahip olduğu bilgiye bağlıdır. Bu hipotetik model bazı yan-kanal bilgisi değeri tahminleri yapmak için kullanılır. Bu tahminler, cihazın gerçek, ölçülmüş yan kanal bilgisi değerleriyle karşılaştırılır. Karşılaştırma işlemi veriler üzerinde bazı istatistiksel metodlar kullanılarak yapılır.

Diferansiyel Güç Analizi (DGA) saldırılarında işlenen veri ile güç tüketimi arasındaki ilişki kurulmaya çalışılır. BGA'dakinin aksine, gürültüyü filtreleyebilmek amacıyla çok sayıda ölçüm yapılır. DGA saldırılarının uygulanabilmesi için gerek koşul, algoritma içerisinde bir veya daha fazla ara değerin, az sayıda anahtar biti ve bilinen giriş veya çıkış verisiyle ifade edilebilmesi ya da en azından korelasyonlu olmasıdır. Bir çok algoritma, bu gerek koşulu sağlamaktadır.

DGA saldırılarının uygulanması, BGA saldırılarına göre oldukça zordur. Ancak BGA saldırılarından çok daha güçlüdürler ve karşı tedbir alınması daha zordur. Ayrıca, incelenen sistemin yapısı hakkında çok fazla bir bilgi sahibi olunmasını gerektirmez.

DGA saldırısının uygulanması için öncelikle gerek koşulu sağlayan bir ara değer belirlenir. Bu ara değer genellikle bir bellek elemanının çıkışı olarak seçilir. Çünkü, bellek elemanlarının konum değiştirmeleri bir saat döngüsünün çok kısa bir diliminde gerçekleşir. Bu nedenle giriş verilerinin yarattığı güç tüketimi farkı (dinamik güç tüketimi şeklinde) çok kısa bir zaman aralığında oluşmakta ve ölçülmesi daha kolay olmaktadır.

Hedeflenen ara değer seçildikten sonra algoritmanın seçilen ara değeri hesaplayan kısmı, genellikle algoritmanın ilk veya son turu, N adet farklı açık veri girişi için aynı gizli anahtar kullanılarak koşturulur [14]. Her açık veri için, k adet örnekten oluşan, karşı düşen güç tüketimi ölçümleri kaydedilir. Bu şekilde $N \times k$ boyutunda bir ölçüm matrisi elde edilir. Matrisin her satırı, kullanılan bir açık veri için elde edilen ölçüm sonuçlarını içerir [14, 23].

Elde edilen ölçüm sonuçlarının değerlendirilmesinde farklı istatistiksel yöntemler kullanılmaktadır. En çok kullanılan iki yöntemden birincisinde, yapılan ölçümlerle giriş verilerinin ilişkisini kurmak için ortalamaların uzaklığı hesabı kullanılırken, ikinci yöntemde korelasyon analizi [14] kullanılmaktadır.

DES Algoritmasının Gerçeklemesine DGA Saldırısı

DES algoritmasının son turundaki S-Kutularının çıkışları DGA saldırısı [6] için uygundur. Bu S-Kutularının çıkışlarını belirlemekte önceki turdan gelen ara değerle birlikte 6 anahtar biti kullanılmaktadır. Ayrıca, çıkış değerinden (kapalı veri) geri dönerek bu S-Kutularının çıkışları hesaplanabilir.

Saldırıyı gerçekleştirebilmek için öncelikle, aynı gizli anahtar kullanılarak, m adet farklı giriş değeri için algoritma m defa koşturulur. Her bir giriş değeri için k adet örnekten oluşan, son tura ait, güç ölçümleri ($T_{1...m}[1 \ldots k]$) ve karşı düşen kapalı veri ($C_{1....m}$) kaydedilir. Kapalı veri değerlerinden de yararlanarak, tüm olası 6-bitlik anahtar (K_s) değerleri için S-Kutusu çıkışları hesaplanır.

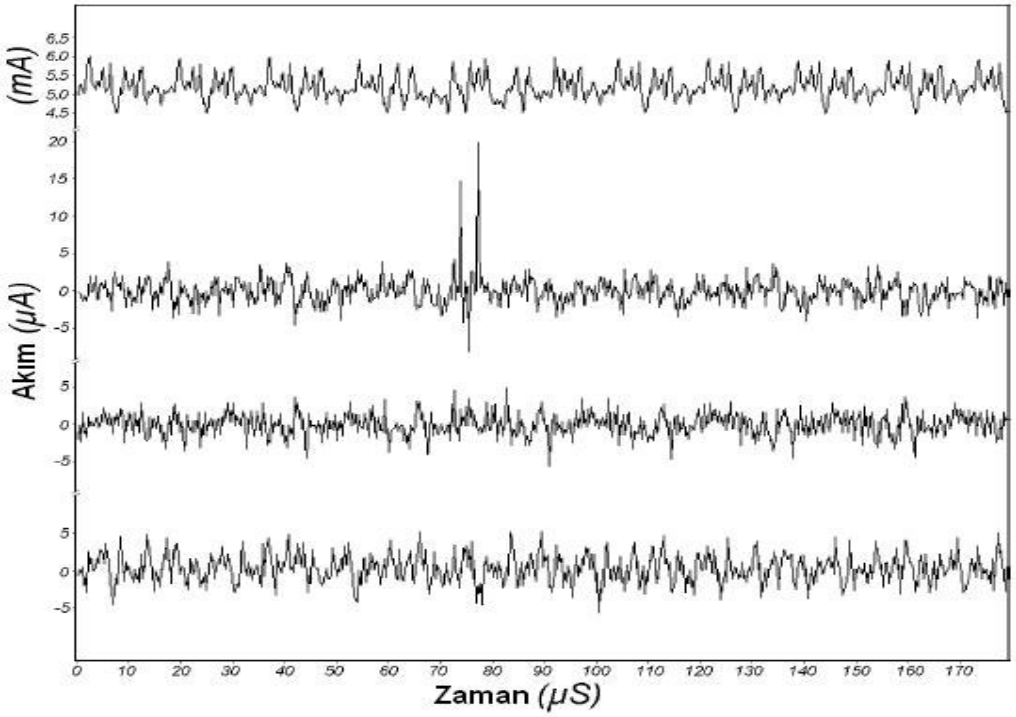
S-Kutusu çıkışlarının belirli sayıda biti dikkate alınarak bir ayrıştırma fonksiyonu oluşturulur. Örneğin bu ayrıştırma, S-Kutusu çıkışının bir tek biti dikkate alıp, bu bitin '1' olarak hesaplandığı giriş verilerine karşı düşen ölçümleri S_1 kümesine, '0' olarak hesaplandığı çıkışları ise S_0 kümesine yerleştirecek şekilde tanımlanabilir. '1' olarak hesaplanan değer, konum değiştirme anında bir $0 \rightarrow 1$ yada $1 \rightarrow 1$ geçişi sonrasında oluşmuştur. $1 \rightarrow 1$ geçişi çok az bir güç tüketimi oluştururken, $0 \rightarrow 1$ geçişleri en yüksek güç tüketimine neden olur. Benzer şekilde, '0' olarak hesaplanan değer, konum değiştirme anında bir $1 \rightarrow 0$ yada $0 \rightarrow 0$ geçişi sonrasında oluşmuştur. $0 \rightarrow 0$ geçişi çok az bir güç tüketimi oluştururken, $1 \rightarrow 0$ geçişleri biraz daha yüksek güç tüketimine neden olur. Bu nedenle S_1 kümesinde daha yüksek güç tüketimine sahip ölçümlerin bulunması beklenir.

Her bir anahtar tahmini için, S_1 ve S_0 kümelerinin ortalamaları arasındaki farktan, ΔT_s ölçüm ortalaması farkı elde edilir. Eğer K_s tahmini yanlışsa, ΔT_s değerinin, ölçüm sayısı artıkça sıfıra yakınsaması beklenir. Çünkü yanlış anahtar tahmini, farklı C_i değerleri için, hesaplanan hedef bit değerinin gerçek değere eşit olması ihtimalini %50'ye indirir. Bu da ayrıştırma fonksiyonunun, hedeflenen bitin

gerçek değeriyle korelasyonsuz hale gelmesine neden olur. Rastgele bir fonksiyon, bir kümeyi iki altkümeye ayırmakta kullanılırsa, yaklaşık olarak eşit elemanlı iki altküme oluşturacağından, bu rastgele oluşturulmuş aynı sayıda elemana sahip iki kümenin ortalamalarının farkının sıfıra yakın olması beklenir.

Eğer K_s tahmini doğruysa, hesaplanan bit değerleri ile gerçek değerler uyumlu olacaktır. Bu da yüksek güç çekilen girişlere karşı düşen ölçümlerin aynı altkümeye toplanmasına neden olacaktır. Bu durumda, ΔT_s değeri, S-Kutusu çıkışlarının hesaplandığı anlarda yüksek değerlere sahip olacaktır.

Şekil 7'de, yapılan üç farklı anahtar tahmini için elde edilen ΔT_s değerlerinin grafikleri verilmiştir. En üstte, DES turuna ait ortalama güç tüketim grafiği bulunmaktadır. Alttaki üç grafikten en üstteki, doğru anahtar tahmini için elde edilen farksal ölçüm değerinin grafiğidir. Görüldüğü gibi, oldukça belirgin tepe değerlerine sahiptir. En alttaki iki grafikte ise yanlış anahtar tahminleri için elde edilmiş olan farksal ölçüm değeri grafikleri görülmektedir.



Şekil 7

3.2.1 DGA Saldırılarına Karşı Tedbirler

DGA saldırılarında, güç tüketimiyle işlenilen veriler, dolayısıyla da gizli bilgiler arasında ilişki kurulur. Bu nedenle, karşı tedbirler, gizli bilgiyle güç tüketimi arasındaki ilişkiyi zayıflatmayı veya güç ölçümlerini zorlaştırmayı hedefler. Karşı tedbirler iki grup altında toplanabilir; yazılımsal karşı tedbirler ve donanımsal karşı tedbirler [24, 25].

Yazılımsal karşı tedbirlerde, işlenen veriyle güç tüketimi arasındaki ilişki zayıflatılmaya çalışılır. İşlem zamanlarının rastgele hale getirilmesi [7] yönteminde, algoritma işlemleri arasına NOP'lar ya da gereksiz (aynı zamanda algoritma işleyişine etki etmeyen) işlemler yerleştirilerek, algoritma işlemlerinin her seferinde farklı anlarda yapılması sağlanmaya çalışılır. Bu şekilde, yapılan ölçümlerde, hedeflenen algoritma parçasındaki güç tüketimleri farklı anlara denk düşer. Böylece ortalama alınarak gürültünün etkisinin azaltılması ve farklı her ölçüm için aynı anda oluşan güç tüketimlerinin üst üste toplanarak yan-kanal bilgisinin güçlendirilmesi mümkün olmaz. Ancak bu yöntemin kullanılması algoritmanın daha uzun sürede tamamlanmasına ve birim zamanda üretimin düşmesine neden olur.

Algoritma işlemlerinin yerlerinin değiştirilmesi [8] ve algoritmanın çalıştırıldığı saat frekansının sürekli olarak değiştirilmesi yöntemlerinde de yukarıda bahsedilen yöntemle aynı amaca ulaşılmaya çalışılır.

Önceki bölümlerde de bahsedildiği gibi, DGA saldırısının gerçekleştirilebilmesi için, algoritma içerisinde sadece anahtar ve bilinen giriş/çıkış verisine bağlı ara değerlerin bulunması gereklidir. Maskeleme yönteminde [9, 10], algoritma girişindeki

veri rastgele bir değişkenle (maske) toplanır. Böylece algoritma içerindeki tüm ara değerler bu üçüncü değere de bağımlı olur ve saldırı için gerek koşul sağlanamaz. Algoritma çıkışında doğru değerin elde edilebilmesi için, algoritma yapısının, bu maske değerinin algoritma işleyişini bozmayacağı şekilde, değiştirilmesi gerekir

Donanımsal karşı tedbirler, yapılan güç ölçümlerini, gürültüyü artırarak ya da sızan yan-kanal bilgisini azaltarak zorlaştırmaya çalışır. Bu tedbirler sonucunda yan-kanal bilgisi sıfırlanamaz. Ancak gerekli ölçüm sayısını pratik olmayan seviyelere çıkarmak mümkün olabilir. Kullanılan bazı donanımsal karşı tedbirler şöyledir; Algoritmayı gerçekleyen devre içerisine bir rastgele sayı üreteci yerleştirilerek [6], ortamdaki gürültü seviyesi arttırılabilir. Ya da güç işaretini filtreleyen bir devre yerleştirilerek [11] yan-kanal bilgisinin genliği oldukça düşürülebilir. Ancak bu yapılar saldırgan tarafından, kurcalamayla, devre dışı bırakılabilir.

4 Elektromanyetik Analiz Saldırıları

Lojik kapıların işledikleri veriye göre güç tüketimi değişmektedir. Bu, çektikleri akımın da değişmesine neden olur. Böylelikle yeni bir yan-kanal bilgisi oluşur. Çünkü devre içerisinde çekilen akımların değişmesi, devrenin yaydığı elektromanyetik radyasyonun da değişmesine, dolayısıyla yan-kanal bilgisi sızdırmasına neden olur. Ayrıca, devre içerisinde oluşan çeşitli kuplajlar ve gerçekleştiren modülasyonlar da elektromanyetik radyasyona, dolayısıyla yan-kanal bilgisi oluşumuna neden olabilir. Elektromanyetik Analiz (EMA: Electromagnetic Analysis) saldırıları bu yan-kanal bilgisini kullanır. Elektromanyetik yan-kanal bilgisini kullanan ilk saldırılar [12] 2000'li yıllarda gerçekleştirilmiş, ardından çalışmalar hızlanmıştır.

EMA saldırıları da basit (SEMA: Single EMA) ve diferansiyel (DEMA: Differential EMA) olarak ikiye ayrılmaktadır. Ölçüm aşamasından sonra, EMA [13] saldırılarında kullanılan analiz yöntemleri, güç analizi yöntemlerinde kullanılanlarla hemen hemen aynıdır.

EMA saldırılarının güç analizi saldırılarına göre önemli avantajları vardır. Öncelikle, GA saldırılarının aksine, ölçümler hedeflenen cihazla hiçbir fiziksel bağlantı kurmadan da belirli bir mesafeden gerçekleştirilebilmektedir. Ayrıca ölçümler cihaz üzerinde istenilen noktalara odaklanabilir.

4.1 EMA Saldırılarına Karşı Tedbirler

TEMPEST standartlarına uyulması ve manyetik radyasyonu engelleyici kaplama yapılması elde edilecek yan-kanal bilgisini zayıflatır. Ancak cihazın saldırganın eline geçmesi durumunda bu tedbirler, koruyucu yapıların çıkarılmasıyla, etkisiz hale getirilebilir. Bunun dışında, GA saldırılarına karşı alınacak tedbirler, EMA saldırılarını da önleyici nitelikte olacaktır.

5 Sonuç

Günümüzde, kriptografik algoritma gerçeklemeleri, sayısal imza, veri şifreleme, güvenli e-posta, finansal transferler, elektronik ticaret ve benzeri bir çok alanda yaygın olarak kullanılmaktadır. Kullanıldıkları tüm sistemlerde güvenlik büyük ölçüde kullanılan kriptografik algoritmalara dayanmaktadır. Bu nedenle kullanılan algoritmaların ve bu algoritmaların gerçeklemelerinin her türlü saldırıya karşı dayanıklı olmaları bir zorunluluktur.

Görece olarak yeni bir konu olan yan-kanal saldırıları, klasik kriptanaliz yöntemlerine karşı dayanıklı olan birçok algoritma için gerçekleme aşamasından sonra büyük bir tehdit oluşturmaktadır. Bu nedenle algoritma gerçeklemelerinde mümkün olduğunca karşı tedbirlerin alınması, algoritma tasarımında ise yan-kanal saldırılarına imkan verecek yapılardan kaçınılması gerekmektedir.

Bu konuda yapılmakta olan akademik çalışmaların yanı sıra, konu ticari olarak da önem taşımaktadır. Yurtdışında özellikle akıllı kart alımlarında yan-kanal saldırılarına karşı güvenilirlik de bir kriter olarak aranmakta ve bu konuda çalışan çeşitli kuruluşlar güvenilirlik sertifikaları vermektedir. Ülkemizde de yapılacak bu tür işlemler için açık bir pazar bulunmaktadır.

Teşekkür

Bu çalışmayı hazırlamama yardımcı olan Sayın Yrd. Doç. Dr. Sıddıka Berna Örs Yalçın'a teşekkürlerimi sunarım.

Kaynaklar

[1] Anderson, R., Kuhn, M., Tamper resistance – a cautionary note, Proceedings of the 2nd USENIX Workshop on Electronic Commerce, 1-11, 1996

[2] Kommerling, O. ve Kuhn, M.G., Design principles for tamper resistant smartcard processors, Workshop on Smartcard Technology 1999

[3] Boneh, D., DeMillo, R.A., Lipton R.J., On the importance of checking cryptographic protocols for faults, EUROCRYPT'97, vol 1233, 37-51, 1997

[4] Kocher, P., Timing attacks on implementations of Diffie-Hellman, RSA, DSS and other systems, CRYPTO'96, vol. 1109, 104-113, 1996

[5] Janke, M. ve Laackmann, P., Power and timing analysis attacks against security controllers. In¯neon Technologies AG, Technology Update, Smart Cards.

[6] Kocher, P., Jaffe, J. ve Jun, B., Differential power analysis, CRYPTO'99, vol. 1666, 388-397, 1999

[7] Chari, S., Jutla C.S., Rao, J.R. ve Rohatgi, P., Towards sound approaches to counteract power-analysis attacks, CRYPTO'99, v1666, 398-412, 1999

[8] Goubin, L. ve Patari, J., DES and differential power analysis the "duolication" method, CHES-1999, vol. 1717, 158-172. 1999

[9] Akkar, M.L. ve Giraud, C., An implementation of DES and AES, secure against some attacks, CHES 2001, Third International Workshop., vol. 2162, 309-318, 2001

[10] Oswald, E., Mangard, S., Pramstaller, N., Rijmen, V., A side-channel analysis resistant description of the AES S-Box, FSE 2005, vol. 3557, 2005

[11] Shamir, A., Protecting smart cards from passive power analysis with detached power supplies, CHES-2000, vol 1965, 71-77, 2000

[12] Quisquater, J. ve Samyde, D., Electromagnetic analysis (EMA): measures and countermeasures fors mart cards, Proceedings of smart card programming and security, LNCS 2140, 200-210, 2001

[13] Chari S., Rao J.R. ve Rotagi P., Advances in Side-Channel Analysis, RSA Laboratories Cryptobytes, vol. 6, 20-32, 2003

[14] Örs, S.B., Hardware Design Of Elliptic Curve Cryptosystems And Side-Channel Attacks. PhD thesis, Katholieke Universiteit Leuven, Faculteit Toegepaste Wetenschappen, Departement Elektrotechniek, Kasteelpark Arenberg 10, 3001 Leuven (Heverlee), Belgium, February 2005.

[15] Joye M., Lenstra A.K. and Quisquater J.-J, Chinese remaindering based cryptosystem in the presence of faults. Journal of Cryptology, 4(12) , 241-245, 1999.

[16] Janke M. and Laackmann P., Power and timing analysis attacks against security controllers. In¯neon Technologies AG, Technology Update, Smart Cards.

[17] Dhem J.F., Design of an effcient public-key cryptographic library for RISC-based smart cards. PhD thesis, UCL Crypto Group, Laboratoire de microelectronique (DICE), May 1998.

[18] Walter C.D., Montgomery exponentiation needs no final subtraction. Electronic letters, 35(21) 1831-1832, October 1999.

[19] Walter C.D., MIST: An efficient, randomized exponentiation algorithm for resisting power analysis. vol 2271 of Lecture Notes in Computer Science, pages 53-66, San Jose, USA, February 2002.

[20] Hachez G. and Quisquater J.-J.. Montgomery exponentiation with no final subtractions: Improved results. In C. K. Koç and C. Paar, editors,Proceedings of 2nd International Workshop on Cryptographic Hardware and Embedded Systems (CHES), vol 1965 pages 293-301, Worcester, Massachusetts, USA, August 17-18 2000.

[21] Kang S.-M., and Leblebici Y., CMOS Digital Integrated Circuits: Analysis and Design. McGraw Hill, 2002.

[22] Ordu L., AES Algoritmasının FPGA Üzerinde Gerçeklenmesi ve Yan-Kanal Analizi Saldırılarına Karşı Güçlendirilmesi. Yüksek Lisans Tezi, İTÜ Fen Bilimleri Enstitüsü, Haziran 2006.

[23] Oswald E., On Side-Channel Attacks and the Application of Algorithmic Countermeasures. PhD Thesis. June 2003.

[24] Messerges T.S., Power Analysis Attacks and Countermeasures on Cryptographic Algorithms. PhD thesis, University of Illinois, 2002.

[25] Borst J., Block Ciphers: Design, Analysis and Side-Channel Analysis. PhD thesis, K.U.Leuven, September 2001.