T.C.
EGE ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

KRİPTOLOJİDE YAN KANAL ANALİZLERİ VE UYGULAMALARI

Yüksek Lisans Tezi

Dursun DEMİROZ

Matematik Anabilim Dalı
Bilgisayar Bilimleri Yüksek Lisans Programı

İzmir
2021

T.C.

EGE ÜNİVERSİTESİ

Fen Bilimleri Enstitüsü

KRİPTOLOJİDE YAN KANAL ANALİZLERİ VE UYGULAMALARI

Dursun DEMİROZ

Danışman: Dr. Öğr. Üyesi Arif GÜRSOY

Matematik Anabilim Dalı
Bilgisayar Bilimleri Yüksek Lisans Programı

İzmir
2021

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ETİK KURALLARA UYGUNLUK BEYANI

E.Ü. Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca Yüksek Lisans Tezi olarak sunduğum "KRİPTOLOJİDE YAN KANAL ANALİZLERİ VE UYGULAMALARI" başlıklı bu tezin kendi çalışmam olduğunu, sunduğum tüm sonuç, doküman, bilgi ve belgeleri bizzat ve bu tez çalışması kapsamında elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara atıf yaptığımı ve bunları kaynaklar listesinde usulüne uygun olarak verdiğimi, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını, bu tezin herhangi bir bölümünü bu üniversite veya diğer bir üniversitede başka bir tez çalışması içinde sunmadığımı, bu tezin planlanmasından yazımına kadar bütün safhalarda bilimsel etik kurallarına uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul edeceğimi beyan ederim.

12/ 07/ 2021

İmzası

Dursun DEMİROZ

ÖZET

KRİPTOLOJİDE YAN KANAL ANALİZLERİ VE UYGULAMALARI

DEMİROZ, Dursun

Yüksek Lisans Tezi, Matematik Anabilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Arif GÜRSOY

Temmuz 2021, 83 sayfa

Bu tezde akıllı kartlarda kullanılan algoritmalar ve bu algoritmalar üzerinde uygulanan yan kanal analizleri incelenmiştir.Yan kanal analizleri, algoritmanın gerçekleştiği esnada istemsiz ürettikleri çıkışları kullanır. Kullanılan bu çıkışlar sisteme ait bilgileri barındırmaktadır. Bu bilgiler sayesinde sisteme ait olan gizli bilginin tamamına ya da bir kısmına ulaşılmasını sağlar.

Akıllı kartlarda kullanılan kriptografik algoritmalar iletişim için istemsiz çıkışlar üretmektedir. Bu istemsiz çıkışların (akıllı kart ile cihaz arasında iletişim sağlayan RFID sinyaller) aktardığı bilgi, kart içinde saklanan gizli bilginin tamamı veya bir parçasıyla ilişkili olduğu için yan-kanal bilgisi olarak tanımlanmaktadır. Yan kanal analizi saldırıları bu yan kanal bilgilerini kullanarak kart içindeki gizli bilginin tamamına ulaşmaya çalışmaktadır. Yan kanal analizi saldırıları, kriptografik algoritmaların kullanıldığı temassız kart sistemleri için büyük bir tehdit oluşturmaktadır. Tez kapsamında temassız kartlara uygulanan yan kanal analiz saldırıları ve uygulamalarının araştırılması hedeflenmiştir.

Anahtar sözcükler: Yan kanal analizleri, Akıllı kart, AES, DES, Güç analizi saldırıları

ABSTRACT

SIDE CHANNEL ANALYSIS AND APPLICATIONS IN CRYPTOLOGY

DEMİROZ, Dursun

MSc in Mathematics

Supervisor: Dr. Öğr. Üyesi Arif GÜRSOY

July 2021, 83 pages

In this thesis, algorithms used in smart cards and side channel analyses applied on these algorithms were examined. Side channel analyses use the outputs produced by the devices used in the realization of the algorithm out of the system. These outputs used contain information about the system. Thanks to this information, it provides access to all or part of the secret key belonging to the system.

Cryptographic algorithms used in smart cards produce leakage for communication. The information transmitted by the leakages (RFID signals that communicate between the smart card and the device) is defined as side channel information since it is related to all or part of the secret key stored in the card. Side channel analysis attacks try to reach all the secret key in the card by using this side channel information. Side channel analysis attacks create a major threat to contactless card systems using cryptographic algorithms. Within the scope of the thesis, it is aimed to investigate side channel analysis attacks and applications applied to contactless cards.

Keywords: Side Channel Analysis, Smartcard, AES, Des, Power analysis Attack

ÖNSÖZ

Kriptoloji geçmişten günümüze gelen bilginin saklanması, korunması ve güvenli bir şekilde iletilmesini sağlayan bir şifreleme bilimidir. Bilgiler kriptoloji sayesinde şifrelenmektedir. Bu bilgilerin çözüm anahtarı bilinmediği sürece bilginin içeriğine ulaşılamaz. Bu sayede tarihte bazı savaşlar kazanılmıştır. Özellikle gelişen teknoloji sayesinde kriptolojinin önemi giderek artmaktadır. Çünkü günlük hayatta kullandığımız çoğu ürün ya da cihazların güvenliği kriptoloji yani şifre bilimi sayesinde sağlanmaktadır. Bu alanda çalışmayı tercih etme sebebim kriptolojinin günlük hayatımızda büyük bir öneme sahip olmasıdır. Ayrıca bir diğer nedeni de ülkemizde bu alanda çalışan kişi sayısının yetersiz olmasıdır.

Yüksek lisans tez konumu belirlerken, literatüre katkı sağlamak ilk hedefim olmuştur. Yaptığım çalışma özellikle pandemi döneminde kullanılmaya başlanan temassız kartlarda kullanılan algoritmalar ve bu algortimaların güvenirliği hakkındadır. Değerli hocalarım Dr. Öğr. Üyesi Arif GÜRSOY, Prof. Dr. Urfat NURİYEV ve Dr. Öğr. Üyesi Erdem ALKIM'ın desteği ile tez konumun belirlenmesi ve yazım sürecinde yaşadığım tüm zorlukların üstesinden geldim. Özellikle tezim boyunca yardımlarını esirgemeyen danışman hocam Dr. Öğr. Üyesi Arif GÜRSOY sonsuz teşekkürü borç bilirim. Ayrıca tez çalışmam boyunca TÜBİTAK-BİDEB 2210-C Öncelikli Alanlara Yönelik Yurt İçi Yüksek Lisans Burs Programı 2019/1 ile maddi destek veren TÜBİTAK'a teşekkürlerimi sunarım.

İZMİR

12/ 07/ 2021 Dursun DEMİROZ

İÇİNDEKİLER

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

ŞEKİLLER DİZİNİ

ŞEKİLLER DİZİNİ (devam)

ŞEKİLLER DİZİNİ (devam)

ŞEKİLLER DİZİNİ (devam)

ŞEKİLLER DİZİNİ (devam)

Şekil Sayfa

TABLOLAR DİZİNİ

KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
AES	Advanced Encryption Standard
CMOS	Complementary Metal Oxide Semiconductor
CPA	Correlation Power Analysis
DEMA	Differential EM Analysis
DES	Data Encrption Standard
DPA	Differential Power Analysis
RSA	Rivest Shamir Adleman
SCA	Side Channel Analysis
SPA	Simple Power Analysis
3DES	Triple Data Encryption Standart

1. GİRİŞ

Kriptoloji günlük yaşantımızın her alanında büyük bir önem taşımaktadır. Özellikle haberleşme alanında çok büyük bir öneme sahiptir. Geçmişte, gönderilen şifreli mesajların çözülmesinden kaynaklı savaşlar kaybedilmiştir. Gelişen teknoloji ile günlük hayatta kullandığımız herşey elektronik ortamda gerçekleşmektedir. Dünya genelinde yaşanan COVID 19 pandemi sürecinde temasın azaltılması için başlatılan uzaktan çalışma sistemine geçilmesiyle kriptolojinin önemi de giderek artmıştır. Yapılan işlemlerin, gönderilen ya da alınan dosyaların, kullanılan temassız akıllı kartların güvenliği insanlar tarafından daha dikkat cekici duruma gelmiştir. Ayrıca COVID 19 pandemi sürecinde kullanımı en çok artan temassız akıllı kartlar günlük yaşantımızı kolaylaştırmaktadır.

Akıllı kartlar günlük yaşantımızda belli yerlerdeki işlemleri yaparken hızlı bir şekilde işlemin gerçekleşmesini sağlamaktadır ve bu sayede akıllı kartlar bizlere zaman kazandırmaktadır. Ayrıca boyutunun ince ve küçük olması da en önemli etkenlerdendir. Fakat, kullandığımız kartlar, içinde bulunan bellekler sayesinde işlem yapmaktadır. Bu belleklerde kartın güvenliğini sağlayan şifreleme algortimaları bulunmaktadır. Bu algortimalar farklı kodlamalarla yazılmaktadır. Akıllı kartlarda kullanılan en yaygın algortimalar AES ve DES şifreleme algoritmalarıdır. Bu algoritmaların tercih edilmesindeki en önemli neden 8 bitlik uzunlukta etkin bir şekilde akıllı karta uygulanmalarıdır. Bu şifreleme algoritmaları AES 128, 192 ve 256 bitlik formatta bulunmaktadır. Ancak bu bit seviyelerindeki algoritmaların hafıza boyutu akıllı kartların belleklerinden yüksek olduğu için uygulanamamaktadır. Böylelikle güvenlik zaafiyeti verilerek daha düşük bitte algoritma uygulanmaktadır.

Bu tezde yan kanal analizleri, akıllı kartlar ve akıllı kartlarda kullanılan algoritmaların yan kanal analiz uygulamaları üzerinde durulmuştur. 2. bölümde kriptoloji ve kriptolojide kullanılan şifreleme algoritmaları anlatılmıştır. 3. bölümde

yan kanal analizleri ve yan kanal analiz saldırıları türlerinin temel çalışma mantığı üzerinde durulmuştur. 4. bölümde akıllı kart, akıllı kart türleri ve akıllı kartlarda kullanılan şifreleme algoritmaları anlatılmıştır. 5. bölümde akıllı kartlarda kullanılan AES ve DES şifreleme algoritmalarına örnek uygulamalar verilmiştir. Bu uygulamalarda akıllı kartlarda kullanılan şifreleme algoritmalarının nasıl deşifre edildiği ve güvenlik anahtarına nasıl ulaşılabileceğine dair yöntemler anlatılmıştır.

2. KRİPTOLOJİ

Geçmişten günümüze kadar olan süreçte her alanda yapılan çalışma ve haberleşmelerin gizliliğine önem verilmiştir. Çalışma ve haberleşmedeki iletilerin başkaları tarafından erişilmemesi için sürekli günün teknolojisine uygun önlemler alınmıştır. Geçmişte savaşlarda ve ülkeler arası haberleşmede kullanılan şifrelenmiş mesajlar köleler aracılığıyla taşımaktaydı. Fakat günümüzde artık bu mesajlar çoğunlukla internet aracılığıyla iletilmekte ve gelişmiş şifreleme sistemleri kullanılmaktadır. Bu sistemlerin asıl amacı iletilmek istenen mesajın farklı harf, sembol ve sayı ile şifrelenip anlaşılmamasını sağlamaktır.

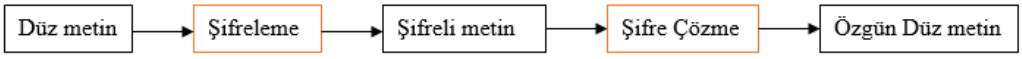
Günümüzdeki yaşam şartları ve koşulları sebebiyle neredeyse herkes interneti bir şekilde aktif olarak kullanmaya başlamıştır. Hatta, internet, eğitim alanında bile yerini almaya başlamıştır. Çünkü sağlanan bu kolaylıklar sayesinde insanlar zaman kazanabilmektedir. Örneğin; bankacılık işlemleri, e-posta, sanal alışveriş vb. birçok alanda kullanarak zaman kazanılmaktadır. Tabi bunların hepsi internet aracılığıyla sağlandığı için internette yapılan işlemlerin güvenliği giderek önem kazanmaktadır. Bu da kriptolojinin günlük hayatımız için ne kadar önemli olduğunu göstermektedir (Çimen et al., 2009).

İlk kriptografik belgeler M.Ö. 1900 yıllarında Mısırlıların yazdığı kitabelerde standart dışı yazım sisteminden oluşan belgelerdir. Fakat kriptografinin kayıtlara geçmiş ilk kullanımı M.Ö. 1500 yılında bir formülün şifrelenmesiyle oluşmuştur. Daha sonra Yunanlıların kriptolojiyi askeri alanda kullanmasıyla şifreleme giderek önem kazanmaya başlamıştır. 1. ve 2. Dünya Savaşlarında büyük etkisi olan şifreleme bu alanda önemli yer almıştır. Giderek zorlaşan şifreleme yöntemlerine karşılık şifre çözme teknikleri de gelişmiştir. Günümüzde halen şifreleme sistemleri

(Kriptografi) ve şifre çözme teknikleri (Kriptoanaliz) gelişmektedir. Kriptografi ve kriptoanaliz kriptolojinin alt bilim dallarıdır.

Kriptoanaliz şifrenin çözülmesini inceleyen daldır. Yani şifrelenmiş olan bir metinin bazı tekniklerle metni açık metin durumuna getirebilmektir. Kriptoanaliz tekniklerini kullanarak işlem yapan analizci herhangi bir algoritma kullanılarak şifreli metinin tamamını ya da bir kısmını çözmeye çalışır.

Bir metin güvenli hale getirilmek için kriptolojiden yararlanarak şifrelenir. Bu olaya şifreleme denir. Şifrelenen metin iletilmek istenilen yere gönderilir. Gelen metin daha önceden bilinen şifre çözme anahtarı ile açık metne dönüştürülür. Bu olaya deşifreleme denir. Bu işlemler Şekil 1'deki gibi gösterilebilir (Keyman ve Yıldırım, 2004).



Şekil 1. Düz metnin şifrelenmesi ve deşifrelenmesi

Kriptolojide şifreleme algoritmaları simetrik şifreleme algoritmaları, asimetirik şifreleme algoritmaları ve hibrit (karma) şifreleme algoritmaları olmak üzere üçe ayrılır (Şahin, 2015).

2.1 Simetrik Şifreleme Algoritmaları

Simetrik şifreleme algoritmalarında yapılan işlem, gönderilmek istenen mesajı gizli anahtar sayesinde şifrelemektir. Fakat gizli anahtarın mesajı gönderen ve mesajı alan kişi tarafından daha önce belirlenmiş olması ya da mesajın yanında

gizli anahtarın da güvenli bir şekilde iletilmesi gerekir. Mesaj alındığı zaman gizli anahtar sayesinde mesaj deşifre edilerek okunabilir duruma getirilebilir.

Simetrik şifreleme algoritmaları günümüzde yaygın bir şekilde kullanılmaktadır. Bunun sebepleri matematiksel olarak işlem sayısının az olması ve şifreleme ve deşifreleme işlemlerinin asimetrik şifreleme algoritmalarına göre oldukça hızlı gerçekleşmesidir. Simetrik şifreleme algoritmaları blok şifreleme ve dizi şifreleme olarak iki türde incelenebilir. Simetrik şifreleme algoritmalarına Veri Şifreleme Standardı (Data Encryption Standard – DES), Gelişmiş Şifreleme Standardı (Advanced Encryption Standard – AES), Blowfish, Üçlü Veri Şifreleme Standardı (Triple Data Encryption Standard - 3DES), Uluslararası Şifreleme Algoritması (International Data Encryption Algorithm - IDEA), Rivest Şifreleme (Rivest Cipher - RC4), MD5 (Message-Digest Algorithm 5) ve SHA (Secure Hash Algorithm – Güvenli Özetleme Algoritması örnek olarak verilebilir.

2.1.1 Blok şifreleme algoritmaları

Blok şifreleme algoritmasında iletilmek istenen mesajda içerik belli blok uzunlukları halinde gizli anahtar aracılığıyla şifrelenir. Belirtilen belli blok uzunlukları bit olarak adlandırılır. Deşifreleme işleminde ise şifreli mesaj gizli anahtar aracılığıyla açık mesaj durumuna getirilir.

Blok şifreleme algoritmaları Shannon'un (Shannon, 1949) karıştırma (confision) ve yayılma (diffusion) teknikleri kullanılarak yapılır. Karıştırma tekniği açık mesaj ile şifreli mesaj arasında bulunan bağı gizlemek için kullanılır. Yayılma tekniği ise açık mesajda yapılan işlemlerin şifreli mesajdaki izinin anlaşılmamasını sağlamak için kullanılır. Karıştırma tekniğinde yer değiştirme, yayılma tekniğinde ise doğrusal dönüşüm kullanılır. Blok şifreleme algoritmalarının gücü anahtar

büyüklüğü, S-kutuları (S-box) ve doğrusal dönüşümler aracılığıyla sağlanır (Aslan et al., 2012).

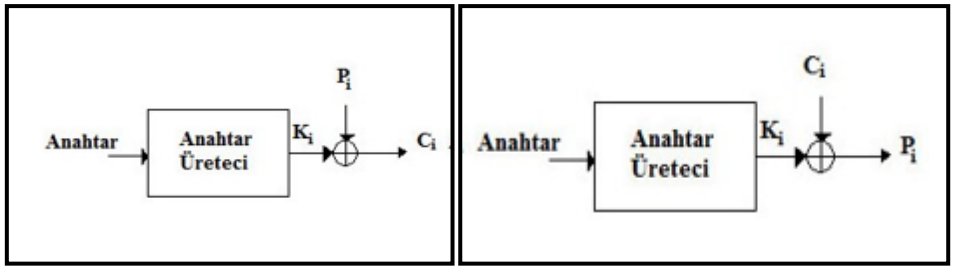
Anahtar büyüklüğü şifrenin kaba kuvvet saldırılarına karşı güçlü olmasını sağlar. En yaygın olan blok şifreleme algoritmalarının anahtar uzunluğu DES (Data Encryption Standard) 56-bit ve AES (Advanced Encryption Standard) ise 128, 192, 256-bit şeklindedir. Kaba kuvvet saldırılarına göre AES DES'e göre daha güçlüdür.

S-box şifrelemenin en önemli aşamasıdır ve karıştırma tekniği burada gerçekleşir. Seçilecek olan S-box'ın iyi olması şifrenin daha karmaşık olmasını sağlar. S-box içerisinde bit blokları yer değiştirerek farklı bit bloklarına haritalanır.

Doğrusal dönüşüm aşamasında ise oluşan giriş bloğu doğrusal yöntemle karıştırılarak aynı uzunlukta çıkış bloğunun elde edilmesi sağlanır.

2.1.2 Dizi Şifreleme Algoritmaları

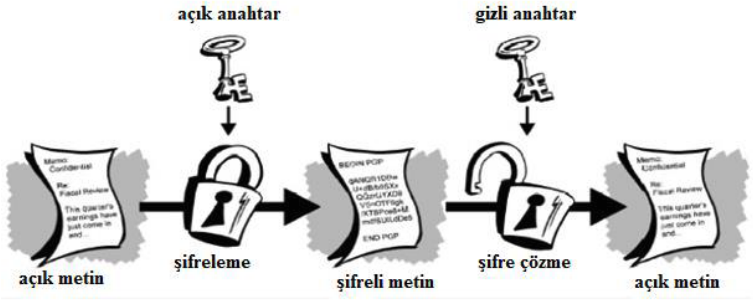
Dizi (Akış) şifreleme algoritmalarında veriyi rastgele üretilen anahtar dizi ile şifreleme işlemi gerçekleştirilir. Deşifreleme aşamasında rastgele üretilen anahtar kullanılarak veriye ulaşılabilir. Dizi şifreleme işlemi blok şifreleme işlemine göre daha hızlıdır ve daha düşük donanım gereksinime ihtiyaç duyar (Ordu, 2006). Fakat dizi şifreleme işleminin güvenlik seviyesi daha düşüktür. Bu şifreleme işleminin genelde 1 kere kullanılması güvenlik açığı riskini azaltmaktadır. Aksi durumda şifrelenen veride güvenlik açığı oluşmaktadır. Şekil 2' de dizi şifreleme işleminin şifreleme ve deşifreleme işlemi gösterilmektedir.



Şekil 2. a) Şifreleme işlemi b) Deşifreleme işlemi.

2.2 Asimetrik Şifreleme Algoritmaları

Asimetrik şifreleme işleminde 2 farklı anahtar kullanılmaktadır. Bir anahtar şifreleme işlemini gerçekleştirmek için diğer anahtar ise şifrelenmiş olan veriyi çözmek içindir. Şifreleme işlemi gerçekleştirilirken açık anahtar (public key), deşifreleme işleminde gizli anahtar (private key) kullanılır. Bu anahtarlar birbirinden bağımsız üretilmesine karşın birbiri arasında matematiksel bağlantı bulunmaktadır.



Şekil 3. Açık Metnin şifrelenmesi ve çözülmesi.

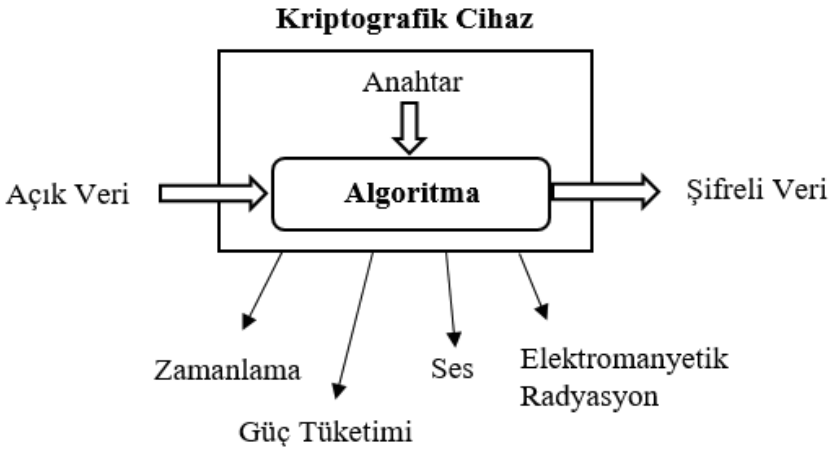
Şekil 3'te görüldüğü gibi gönderici, açık anahtar ile şifreleme işlemini yapmaktadır. Alıcı ise elinde bulunan gizli anahtar ile şifreli metni açık metne dönüştürmektedir. Asimetrik şifrelemede açık anahtarı gizlemeye gerek yoktur. Fakat asimetrik şifreleme simetrik şifreleme işlemine göre daha yavaştır (Hatun, 2018). Bunun sebebi şifrelerin uzun olması yani bit uzunluğundan kaynaklanmaktadır. Bit uzunluğunun uzun olması asimetrik şifrelemenin daha güvenilir olmasını sağlamaktadır. Asimetrik şifreleme algoritmalarına Diffie Helman, RSA(R.Roland, A.Shamir, L. Adleman), DSA (Digital Signature Algorithm) ve ECC (Eliptik Eğri Algoritması) örnek olarak verilebilir.

2.3 Hibrit Şifreleme Algoritmaları

Günümüzde kullanılan melez sistem olarak adlandırılan bir şifreleme sistemidir. Burada, şifreleme işleminin aşamaların bir kısmı simetrik şifreleme ile bir kısmı asimetrik şifreleme kullanılarak yapılmaktadır. Anahtar şifreleme, anahtar anlaşma ve sayısal imza işlemleri asimetrik şifrelemeyle, veri işlemleri ve veri bütünlüğü korunması da simetrik şifreleme sistemi ile gerçekleştirilir. Böylelikle şifreleme işlemi hem hızlı hem de güvenli bir şekilde gerçekleştirilir. Hibrit şifreleme algoritmasına PGP (Pretty Good Privacy) örnek olarak verilebilir (Yıldırım ve Demiray, 2008).

3. YAN KANAL ANALİZLERİ

Algoritmanın gerçekleşmesinde kullanılan cihazların istem dışı ürettikleri çıkışlar bulunmaktadır. Bu istem dışı çıkışlara yan kanal bilgisi adı verilir. Bu çıkışların sebepleri kullanılan cihazın fiziksel özelliklerinden (tükettiği güç, elektromanyetik dalga, çıkardığı ses, işlem süresi, sıcaklık) kaynaklanmaktadır (Büyükkaya, 2017). Bu çıkışlar yüzünden dışarıya işlem hakkında bilgi sızdırılmasından dolayı sistemsel olarak açıklıklar ortaya çıkmaktadır(Şekil 4). Yan kanal analizlerinde ise bu sızdırılan bilgiler kullanılarak cihaza veya sisteme ait gizli bilginin tamamına ya da bir kısmına ulaşılmasını sağlanır.



Şekil 4. Yan kanal bilgisi.

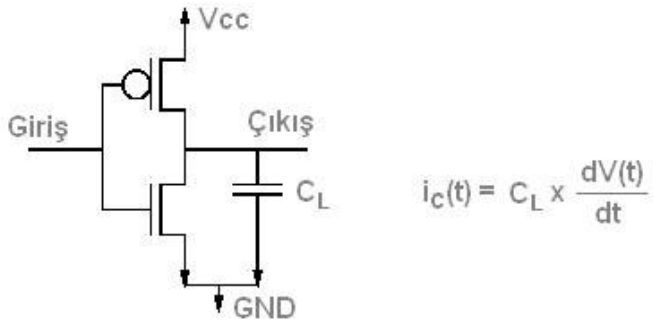
Yan kanal analizleri kendi içinde aktif saldırılar ve pasif saldırılar olmak üzere ikiye ayrılır. Aktif saldırılar doğrudan cihaza yapılan müdahalelerdir. Bu saldırının amacı direk olarak devreye ulaşmaktır. Bu saldırılar uzun süre isteyen çalışmalar gerektirmektedir. Ayrıca aktif saldırılar sonucunda saldırının yapıldığına dair izler kalmaktadır. Bu saldırıları yapılması için de özel cihazlara gerek

duyulmaktadır. Bunlar lazer istasyonları, Focused Ion Beam (FIB) yonga soyma gibi cihazlarıdır. Aktif saldırılar pasif saldırılara göre daha zor ve pahalıdır (Hatun, 2018).

Pasif saldırılar aktif saldırılar gibi devreye veya cihaza ulaşılmadan sistemin çalışma esnasında sızdırdığı yan kanal bilgilerini kullanır. Sızdırılan bu bilgiler sayesinde gizli anahtara ulaşılabilir. Bu tür saldırılarda aktif saldırılardaki gibi pahalı cihazlara gerek yoktur. Pasif saldırılarda hem yazılım hem de işlemci hakkında detaylı bilgiye sahip olmak gerekir. Ayrıca pasif saldırılar aktif saldırılardan daha tehlikelidir. Çünkü pasif saldırılar fark edilmeyen saldırılardır. Pasif saldırılar yan kanal bilgisine göre güç analizi atakları, elektromanyetik analiz saldırıları, zamanlama analizi saldırıları ve akustik analiz saldırıları olmak üzere 4 ayrı grupta incelenir (Ordu ve Örs Yalçın, 2021).

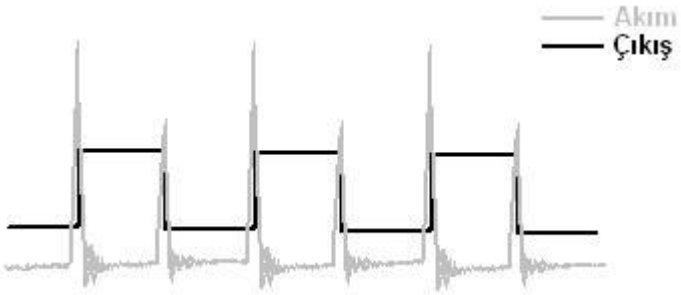
3.1 Güç Analizi Saldırıları

Tamamlayıcı metal oksit yarı iletken (Complementary metal oxide semiconductor (CMOS)) (Şekil 5) günümüz teknolojisinde kullanılan bütünleşmiş devrelerdir. Bu devrelerin düşük enerji tüketimi, geniş gürültü marjları ve kolay tasarlanabilir olmasından dolayı CMOS bütünleşmiş devreler belleklerde, mikroişlemcilerde, sinyal işlemcilerinde vb. alanlarda kullanılmaktadır. CMOS devresindeki transistörün güç tüketimi devrenin güç tüketimine bağlıdır. CMOS transistörünün elektrik yüklerinin hareketi (Şekil 6), anlık gerilim değişimine bağlıdır. Transistör devrelerde anahtar görevi görmektedir. Transistör 0 (sıfır) durumunda ise akım devre dışı kalır 1 (bir) durumunda ise akım geçişi olur. Trasnsistörün 0 olduğu durumlardaki güç tüketimi çok düşük kalmaktadır. Şekil 6'da görüldüğü gibi 0-1 geçişindeki güç tüketimi, 1-0 geçişindekinden daha yüksektir. Böylece geçişler anındaki güç tüketimleri farklarından entegre devrenin güç tüketimi takip edilebilir.



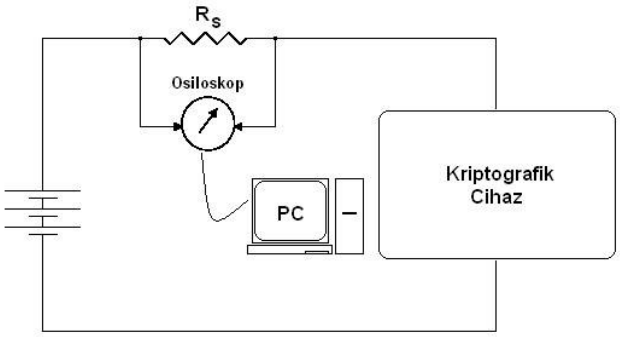
Vcc = Kollektör besleme voltajı CL = Kondansatör GND = Topraklama

Şekil 5. CMOS transistörlerle gerçekleşmiş bir evirici ve yük kapasitesi.



Şekil 6. CMOS çıkışı ve güç tüketimi.

Bu güç tüketiminin analiz edilmesiyle gizli bilgiye ulaşılabilir. Bu bilgiye ulaşabilmek için bütünleşmiş devrenin güç tüketiminin ölçülmesi gerekir. Şekil 7'de gösterildiği gibi kriptografik cihaz devresini besleyen hat üzerine düşük değerli bir direnç bağlanır ve bu direncin iki uç arasındaki gerilim farkını ölçmek için osiloskop bağlanır. Ölçülen gerilim farklarından yararlanılarak değişen akım bilgisine ulaşılabilir.



R_s = Direnç

Şekil 7. Güç analizi şeması.

Güç analizi saldırıları ilk kez Kocher (Kocher et al., 1999) tarafından DES algoritması üzerinde uygulanmıştır. Bu uygulamadan sonra birçok uygulama gerçekleşmiştir.

Güç analizi saldırıları üç başlık altında toplanabilir; basit güç analizi saldırıları, diferansiyel güç analizi ve korelasyon güç analizi saldırılarıdır (Mangard et al., 2007).

3.1.1 Basit güç analizi

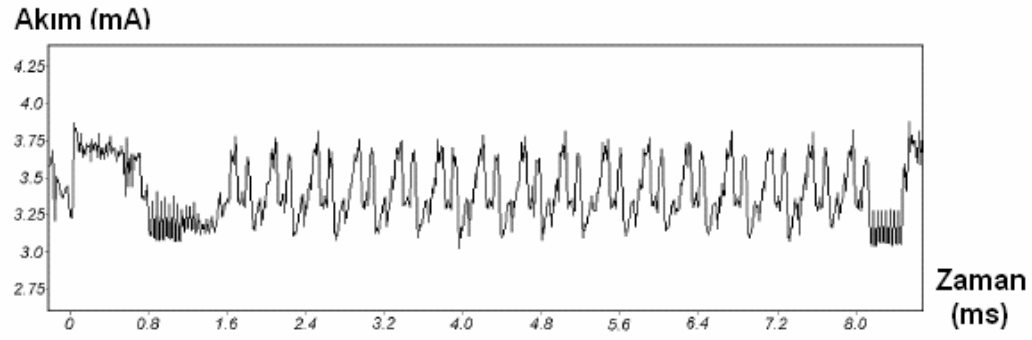
Basit güç analizi saldırısı (Simply Power Analysis (SPA)) ilk olarak Paul Kocher (Kocher et al., 1999) tarafından uygulanıp tanıtılmıştır. Basit güç analizi saldırısı kısa metinlerde kullanılmaktadır ve saldırı için bir veya birkaç güç izi yeterli olmaktadır. Basit güç analizlerinde kripto cihazı çalışırken elde edilen güç izleri gözle analiz edilir ve yorumlanır. Bu analizlerin kullanılabilmesi için saldırıyı yapan kişinin kripto cihaz hakkında detaylı bilgiye ihtiyacı vardır. Bilinen bu

bilgiler ve yapılan analizler sonucunda kripto cihazın hangi algoritmayı çalıştırdığına ve gizli anahtar bilgisine de ulaşılabilir.

Kripto cihaz verilen algoritmadaki aritmetik talimatlardan (toplama gibi), mantıksal talimatlardan, veri aktarım talimatlarından (taşıma gibi) ve dallanma talimatlarından (atlama gibi) oluşan işlemleri gerçekleştirdiği zamanlardaki güç tüketimleri farklıdır. Bu güç tüketimindeki farklılıklar sayesinde kripto cihazın güç izlerindeki farklılık rahatlıkla gözlemlenebilir.

Gözlemler yapılırken güç tüketiminde dikkat edilen iki önemli bilgi vardır. Bunlar Hamming ağırlık bilgisi ve Hamming uzaklık bilgisidir. Hamming ağırlık bilgisi alınan güç izlerindeki "1" bitlerin sayısını, Hamming uzaklık bilgisi ise güç izindeki 0-1 ve 1-0 olan geçişleri tanımlamaktadır.

İlk uygulanan basit güç analizi saldırısı Kocker, Jaffe ve Jun tarafından DES'in 16 döngüsünün güç tüketimini kriptografik işlem sonucunda net bir şekilde elde etmiştir. Şekil 8'de DES algoritmasını çalıştıran akıllı bir karta ait güç tüketimi görülmektedir (Kocher, 1996).



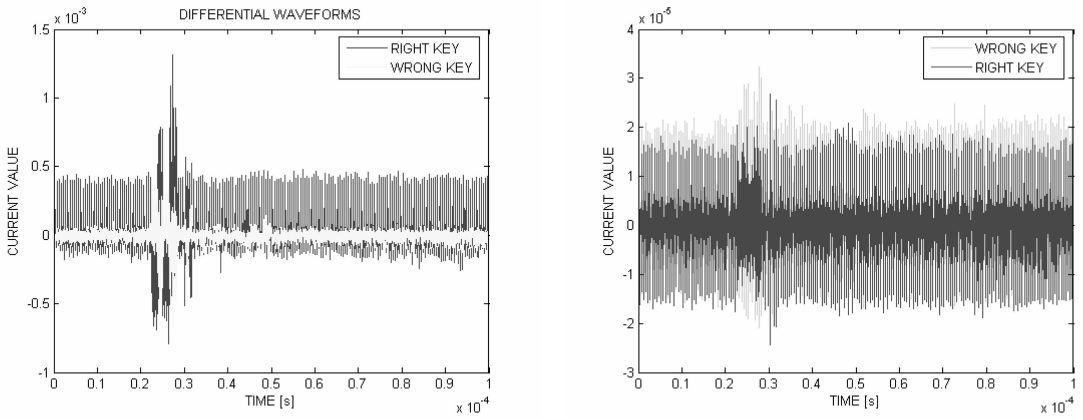
Şekil 8. DES algoritmasını çalıştıran Akıllı Kartın güç ölçümü.

Şekil 8'de görüldüğü gibi DES algoritması çalıştıran akıllı karta ait güç tüketiminde farklılıklar bulunmaktadır. Bu farklılıklar sayesinde akıllı karta ait olan gizli bilgiye ulaşmak mümkündür. Basit güç analizi saldırısını önlemek için koşullu dallanmalardan kaçınılmalı veya saldıran kişi yanıltmak için sahte dallanmalar gerçekleştirilebilir. Bir diğer önlem yöntemi de güç tüketimini dengede tutmaktır. Bunun için algoritmada gereksiz işlemler gerçekleştirilerek güç tüketimini dengede kalmasıyla saldırıyı yapacak kişinin önüne geçilebilir. (Aydoğan, 2016).

3.1.2 Diferansiyel güç analizi

Diferansiyel güç analizleri (Differantial Power Analysis (DPA)) güç analizlerinde en çok kullanılan analiz türüdür. Çünkü basit güç analizinde saldırıya uğrayan cihazın algoritmasında detaylı bilgiye ihtiyaç varken DPA saldırılarında ayrıntılı bilgiye ihtiyaç yoktur. Ayrıca DPA saldırılarında gürültü olsa bile cihaza ait olan gizli anahtar ortaya çıkartılabilmektedir. DPA saldırıları SPA saldırılarına göre daha çok güç izine ihtiyaç duymaktadır. Bunun içinde DPA saldırısını gerçekleştirmek için fiziksel olarak kriptografik cihaza sahip olmak gerekir.

DPA saldırılarının amacı, kriptografik cihazların şifrelerini çözmek için cihazın farklı veri bloklarını şifrelerken ve şifresini çözerken kaydedilen çok sayıdaki güç izlerini kullanarak çözmektir. SPA ile DPA arasındaki en önemli fark alınan güç izlerinin farklı şekillerde analiz edilmesidir. DPA saldırılarında alınan güç izlerinin (Şekil 9) verilere nasıl bağlı olduğunu analiz etmektedir. Yani DPA da alınan güç izleri verilerle ilişkilendirilir.



(a) (b)

Şekil 9. Diferansiyel güç analizi izlerinden örnek.

DPA saldırıları ile kriptografik cihazlarını şifrelerini çözerken kullanılan genel saldırı adımları vardır. Bunlar

Yürütülen Algoritmanın Ara Sonucunu Seçme

Güç Tüketiminin Ölçülmesi

Varsayımsal Ara Değerlerin Hesaplanması

Ara Değerleri Güç Tüketimi Değerleriyle Eşleştirme

Varsayımsal Güç Tüketim Değerlerinin Güç İzleriyle Karşılaştırılması

adımlarıdır.

3.1.2.1 Yürütülen Algoritmanın Ara Sonucunu Seçme

DPA saldırılarındaki ilk adımda kriptografik cihaza ait olan algoritmanın bir ara sonucunu seçmektir. Bu sonuç içerisinde $f(d,k)$ fonksiyonu olması gerekir. Burada d, genellikle düz metin ya da şifreli metindir. k ise anahtara ait küçük bir parçadır. Bu ara sonucun amacı k değerini belirlemektir.

3.1.2.2 Güç tüketiminin ölçülmesi

DPA saldırılarının ikinci adımında kriptografik cihazın farklı D sayısı kadar veri bloklarını şifrelerken ya da şifresini çözerken güç tüketimini ölçmektir. Bu ölçümler sırasında yapılan şifreleme veya şifre çözme işlemi sırasındaki seçilen ara sonucun 1. aşamada belirtilen veri değeri *d* 'nin bilinmesi gerekir. Bu veri değerleri $d = (d_1, \dots, d_D)$ şeklinde yazılır. Burada d_D veri değeri D'ninci şifreleme veya şifre çözme işleminde elde edilen ara sonuç veri değeridir.

Yapılan her saldırı için saldırgan güç izini kaydeder. d_i veri bloğuna karşılık gelen $t = (t_1, \dots, t_i)$ şeklinde güç izleri uzunlukları tanımlanır. Güç izleri kayıt edilirken tüm izlerin doğru şekilde hizalanması gerekir. Bunun içinde güç izi alınırken osiloskobun tetikleme sinyalinin, her şifre çözme ve şifreleme aşamasında aynı işlem sırasında güç tüketimini kaydedecek şekilde ayarlaması gerekir.

3.1.2.3 Varsayımsal ara değerlerin hesaplanması

DPA saldırısının üçüncü aşamasında saldırgan olası her anahtar hipotezi (k) için ara değer hesaplamasıdır. Bu ihtimaller $k = (k_1, \dots, k_K)$ olarak yazılır. Burada K, k değerinin alabileceği ihtimallerin sayısıdır. Alınan değerlerden tüm veri değeri

D ile anahtar hipotezi *K* ihtimalleri sayesinde varsayımsal ara değerleri $f = (d, k)$ fonksiyonu hesaplanabilir. Bu hesaplamalarda DxK boyutunda bir *V* matrisi elde edilir. Denklem 1'de varsayımsal ara değerin hesaplanması verilmektedir.

$$V_{i,j} = f\left(d_i, k_j\right) \;\; i = 1, \dots, D \;\; j = 1, \dots, K \qquad (1)$$

V matrsinin *j* sütunu, k_j anahtar hipotezine göre hesaplanan ara sonuçları gösterir. Her *k* anahtar hipotezi için k kadar olası bir hipotez vardır. *V* sütününda, D bloğunun şifreleme ve şifre çözme çalışmalarında cihaz tarafından hesaplanan tüm ara değerler mevcuttur. Cihazda hesaplanan değerler k anahtar hipotezinin bir öğesidir. Bu öğenin indeksini ck olarak tanımlanır. Bu durumda k_{ck}'ın cihazın anahtarı olarak tanımlanır. DPA saldırısının amacıda D şifreleme ve şifre çözme çalışmasında k_{ck}'nın hangi V sütununda işlendiğini bulmaktır. V sütunu belirlendikten sonra k_{ck} değeri öğrenilir.

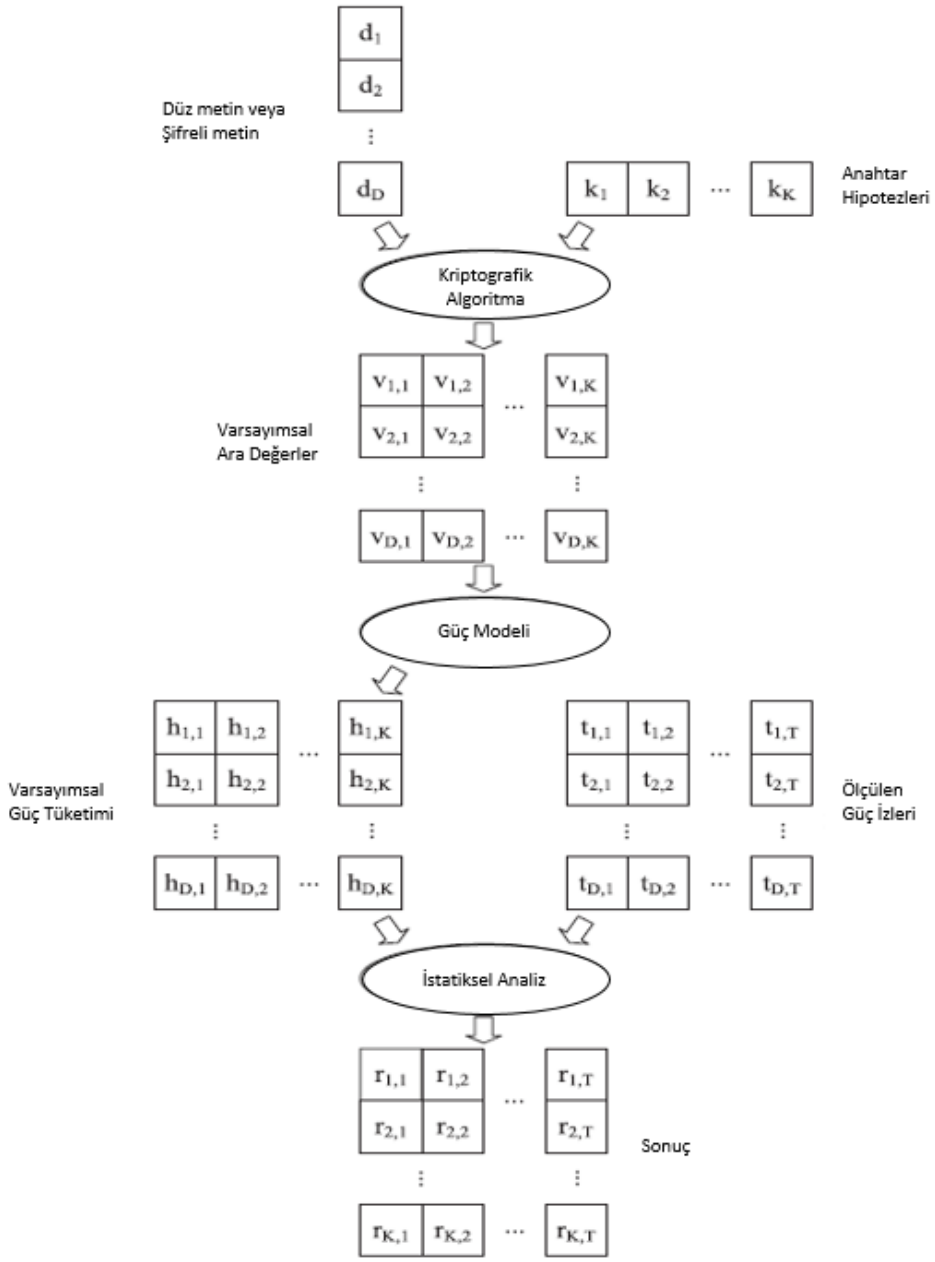
3.1.2.4 Ara değerleri güç tüketimi değerleriyle eşleştirme

DPA saldırısının dördüncü adımında ise, varsayımsal ara değerlerleri *V* varsayımsal güç tüketimi değerlerinin bir *H* matrisine eşlemektir. Bunun için saldırgan güç tüketimi modellerinden bir simülasyon tekniğini kullanır. Kullanılan bu teknik sayesinde *V* varsayımsal güç tüketimi değeri $h_{i,j}$ değerini elde etmek için varsayımsal ara değer $V_{i,j}$ için cihaz güç tüketimini simüle edilir. Yapılan simülasyonun kalitesi saldırganın saldırdığı cihaz hakkındaki bilgisine bağlıdır. Eğer elde edilen simülasyon cihazın gerçek güç tüketimine özelliklerine ne kadar benzer ise yapılan DPA saldırısı o kadar etkili olur. *V* ile *H* değerlerini eşlemek için

kullanılan en yaygın güç modelleri SPA da olduğu gibi Hamming mesafesi ve Hamming ağırlık modelidir.

3.1.2.5 Varsayımsal güç tüketim değerlerinin güç izleriyle karşılaştırılması

DPA saldırasının son adımı (Şekil 10) olarak her anahtar hipotezinin varsayımsal güç değerlerini elde edilen güç izleriyle karşılaştırmasıdır. Yani her bir h, H matrisinin her biri t (alınan güç izi) ile karşılaştırılır. Bu karşılaştırmalar sonucunda her r_j elemanın h_i ve t_j sütunları arasındaki karşılaştırlardan elde edilen KxT boyutunda bir R matrisi oluşur. Algortimaların $r_{i,j}$ değeri ne kadar yüksek ise h_i ile t_j eşleşmesi daha iyi olur.



Şekil 10. DPA saldırısının 3 ile 5 arasındaki adımlarını gösteren blok diyagram.

Güç izleri cihazın yürütttüğü kriptolojik algoritmaya farklı veri girişi yapılarak alınanan güç tüketimine karşılık gelir. İlk adımdaki ara sonuç cihaz tarafından yürütülen algoritmanın bir parçası olduğu için algoritmanın farklı veri girişleri sırasındaki ara değerini (V_{ck}) hesaplanması gerekir. t_{ck} sütunu V_{ck} bağlı olarak hesaplanan güç tüketimi değerlerini içerir. Varsayımsal güç tüketimi (H_{ck}) değerleri V_{ck} değerine bağlı olarak simule edildiği için H_{ck} ile t_{ck}sütunları birbiriyle ilişkilidir. H_{ck} ile t_{ct} değerleri *R* matrisindeki en yüksek değeri gösterir. Bu da *R* matrisindeki $r_{ck,ct}$ değerini karşılık gelir. *R* matrisindeki diğer tüm değerler düşüktür. Çünkü o değerlerde H ile T arasındaki güçlü bir ilişki yoktur. Böylece saldırgan *R* matrisindeki en yüksek değeri arayarak doğru anahtar ck için indeksi ct zaman anını bulabilir.(Mangard et al., 2007).

3.1.3 Korelasyon güç analizi

Korelasyon güç analizi (Correlation Power Analysis (CPA)) diğer analiz yöntemlerine göre daha az güç izine ihtiyaç duymaktadır. Çünkü istatistiksel bir saldırı türüdür. Güç izinden elde edilen verileri ilişkilendirmek için Pearson korelasyon katsayısını kullanır. Saldırının ilk adımında şifreleme veya şifre çözme aşamasında üretilen bir ara değer ve anahtarın bir parçası seçilmektedir.

$$I = f(d, k) \qquad (2)$$

$$I = Ara\ değer$$

$$d = Düz\ metin$$

$$k = Alt\ Anahtar\ (\ Anahtarın\ bir\ parçası)$$

Analizin bir sonraki aşamasında güç ölçümü yapılmaktadır. Burada d metni için duruma bağlı olarak binlerce güç izi elde edilmektedir. Bu güç izleri gerçek güç tüketimi olarak adlandırılmaktadır. N sayıda güç izi elde etmek için N sayıda düz metin kullanılmaktadır. Her güç izi, zaman içinde örneklenen her andaki güç tüketimine karşılık gelen M sayıda örnekleme noktasına sahip olacaktır. Kullanılan her bir değişken veri örneklemesi için ara değer ve bu ara değerlerin güç tüketimi, Hamming mesafe modeli gibi bir güç modeli kullanılarak hesaplama yapılmaktadır. Bu hesaplamada alt anaharın olma ihtimaline karşı tüm olası değerleri için tekrarlanmaktadır. Hesaplanan bu değerleri varsayımsal güç tüketimi değeri olarak adlandırılmaktadır.

Yapılan hesaplamalardan sonra alt anahtarın en yüksek korelasyona ship olduğunu bulmak için ölçüm yapılan her alt anahtarın gerçek güç tüketimi değerleri ile varsayımsal güç tüketimi değerleri karşılaştırılmaktadır. Bu karşılastırma Denklem 2 de verilen Pearson korelasyon katsayısı adı verilen istatiksel yöntem kullanılarak yapılmaktadır. Bu denklem belirli bir alt anahtarının j. örnekleme noktasının tahmini korelasyonunu bulmak için kullanılmaktadır.

$$\hat{\rho} = \frac{N \sum_{i=0}^{N} W_{i,j} H_i - \sum_{i=0}^{N} W_{i,j} \sum_{i=0}^{N} H_i}{\sqrt{\sum_{i=0}^{N} W_{i,j}^2 - \left(\sum_{i=0}^{N} W_{i,j}\right)^2} - \sqrt{\sum_{i=0}^{N} H_i^2 - \left(\sum_{i=0}^{N} H_i\right)^2}} \tag{3}$$

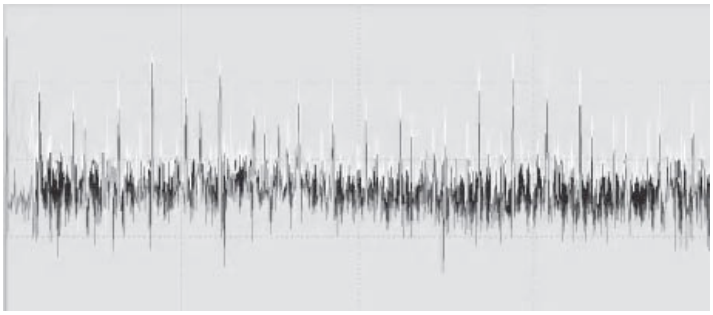
$$W_{i,j} = i = g\text{üç } izi\ j = \text{örnek noktası}$$

$$H_i = \text{Alt anahtara göre i. düz metin için varsayımsal güç tüketimi}$$

Pearson korelasyon katsayısı -1 ile 1 arasında değer almaktadır. Eğer korelasyon katsayısı 1 olursa varsayımsal güç tüketim değerleri ile gerçek güç tüketimi değerleri en iyi korelasyona sahip olduğu anlamına gelmektedir. 0 ise bu iki değer arasında değerler arasında herhangi bir korelasyon olmadığı anlamına gelmektedir. Sonuç -1 çıkar ise en iyi ihtimalle korelasyonlu oldukları fakat değerlerin birbiriyle ters orantılı olduğu anlamına gelir. Böylelikle korelasyon katsayısı bu iki değerin birbiriyle ne kadar ilişkili olduğunu göstermektedir. Bu yüzden varsayımsal güç tüketimi değerleri doğru anahtar kullanılarak hesaplanması maksimum korelasyonun elde edilmesini sağlayacaktır (Gamaarachchi and Ganegoda, 2018).

3.2 Elektromanyetik Analiz Saldırıları

Elektromanyetik analiz saldırıları (Electromanyetik Attack (EMA)) günümüzde kullanılan elektronik devrelerin çoğunda bulunan metal oksitli yarı iletken transistörlar (CMOS) kullanılmaktadır (Kang and Leblebici, 1983). Kullanılma sebeplerinden biri cihaz çalışmadığı durumlarda güç tüketiminin az olmasıdır. Cihaz çalıştığında ise veri girişine bağlı olarak cihazın güç tüketiminde farklılık göstermektedir. Güç tüketiminin farklı olmasından dolayı devredeki akım geçişi de farklı olmaktadır. Akımın değişimi de yayılan elektromanyetik radyasyonun değişmesine sebep olur (Tiu, 2005). Yayılan elektromanyetik radyasyonun değişiminden dolayı devre, yan kanal bilgisinin oluşumunu sağlamaktadır. Şekil 11'de örnek bir elektromanyetik saldırı sinyali gösterilmektedir.



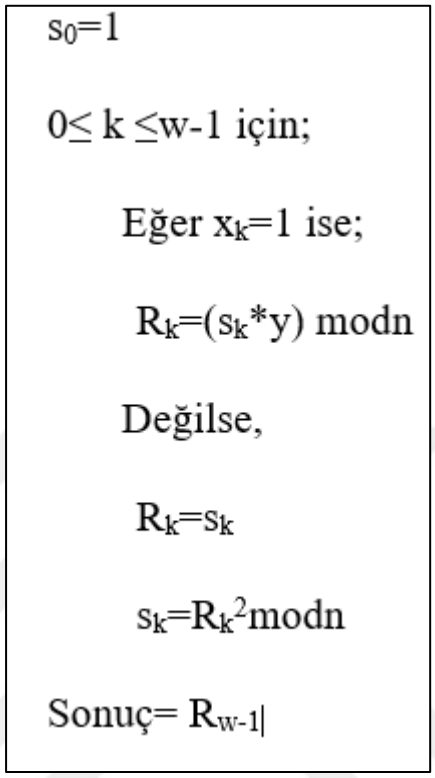
Şekil 11. Alınan bir elektromanyetik saldırı sinyali.

Elektromanyetik analiz saldırılarında güç saldırıları gibi cihaz ya da devre ile temas ya da bağlı olmasına gerek yoktur. EMA saldırılarında kurulan elektromanyetik alıcılar (Şekil 12) sayesinde belirli bir mesafeden ölçüm alınabilmektedir. Güç analiz saldırılarına göre de EMA saldırılarında gürültü ölçümlere etki etmektedir. Gürültü EMA saldırılarında bir sorundur (Le et al., 2007).

Şekil 12. Elektromanyetik Alıcı.

3.3 Zamanlama Analizi Saldırıları

İlk zamanlama analizi saldırısını Kocker 1996 yılında RSA algoritmasını çalıştıran bir akıllı kartı kullanarak gerçekleştirmiştir. Kriptografik algoritmalar işlemlerini gerçekleştirirken farklı zaman tüketmesinden kaynaklanmaktadır. Böylece algoritmada geçekleştirilen işlemin gerçekleşme süresinin farklı olmasından dolayı gizli anahtar ile ilgili yan kanal bilgisi oluşturmaktadır (Kocher, 1996). Örneğin algoritma içerindeki toplama ve çarpma işlemleri farklı sürelerde yapılmaktadır. *x, y, m* bitlik değişken olarak kabul edilirse, $z = x + y$ ve $z = x * y$ işlemlerini hesaplandığını varsayalım. Eğer toplama işlemi $T_{\mathrm{t}} = m$ sürede gerçekleştirir ise çarpma işlemi için toplama işlemini taban alırsak $T_{\mathrm{ç}} = \frac{3x(m-1)xm}{2}$ sürede geçekleşir. Örnekte görüldüğü gibi aynı bitteki iki terimin için yapılan işlemler farklı sürelerde gerçekleşmektedir. Böylece algoritmanın hangi işlemi yaptığı kullanılan süreden tespit edilebilir (Aydoğan, 2016). Kocker zamanlama analizi saldırısını yaptığı çalışma da RSA algoritmasında kullanılan işlem $\mathrm{R} = \mathrm{f(x, y)} = y^x mod(n)$ şeklindedir. Bu işlem de *n* değeri bilinen değer, *y* değeri ise giriş değeridir. Bu durumda saldırganın bulması gereken değer *x* değeridir. Kocker bu çalışmasında modülo üs alıcılarının analizini tanımlamaktadır (Kocher, 1996). Şekil 13 te RSA algortimasına ait $\mathrm{f(x, y)} = y^x mod(n)$ değerini hesaplayan modülo üs algoritması verilmektedir.



Şekil 13. Modülo üs algoritması.

4. AKILLI KARTLAR TEKNOLOJİSİ

Elektronik veri işleme sistemlerindeki gelişmeler sayesinde akıllı kartların gelişmesinde büyük bir katkı sağlamıştı. 1970'lerdeki mikroelektronikteki gelişmeler ile veri depolama ve işleme sisteminin bütünleşmiş yongada muhafaza edilmesini sağladı. 1968 yılında Alman bilim adamları Jürgen Dethloff ve Helmut Grötrupp bütünleşmiş yonga devresini kimlik karta dahil etme fikri üzerinde patent başvurusunda bulundu. Fakat akıllı kartlardaki ilk gerçek çalışma 1974 yılında Roland Moreno'nun akıllı kart patentini Fransa'da kayıt ettirmesiyle gerçekleşti. Zamanın şartlarında üretilen akıllı kartların güvenirliği için birçok teknik sorunlar çözüldü. 1984 yılında Fransa PTT'si telefon kartlarıyla denemesinde başarı elde etti. Bu başarıda akıllı kartların güvenirliği de başarılı bir şekilde sağlandığı görüldü. 1984-85 te ise Almanya'da telefon kartları için farklı kart tiplerinden pilot bir çalışma gerçekleştirdi. Bu çalışmalar sonucunda telefon kartları tüm dünyaya yayılmaya başladı. 1997 yılında dünya üzerinde birkaç yüz milyon kart kullanımdaydı.

Telefon kartı üzerinde bulunan bütünleşmiş devreler, dışarıdan gelecek saldırılara karşı koruma sağlarken kart bakiyesinin azalmasını sağlayan küçük bir yongadır. Gelişen teknoloji sayesinde telefon ile haberleşme uygulamalarında daha büyük ve karmaşık yapıda mikroişlemci yongaları kullanılmaya başlandı. Alman Postanesi 1988 yılında EEPROM (Electronically Erasable Programmable Read-Only Memory) teknolojisini kullanan bir mikroişlemci kartı analog mobil telefon ağ sisteminde kullanarak bu alanda liderlik üstlendi. Fakat mikroişlemci kartların abone sayısında sınırlı kalmasından dolayı yeteri kadar etki göstermedi. Bundan dolayı akıllı kartlar dijital GSM ağına dâhil edilerek 1991 yılında hizmete girdi.

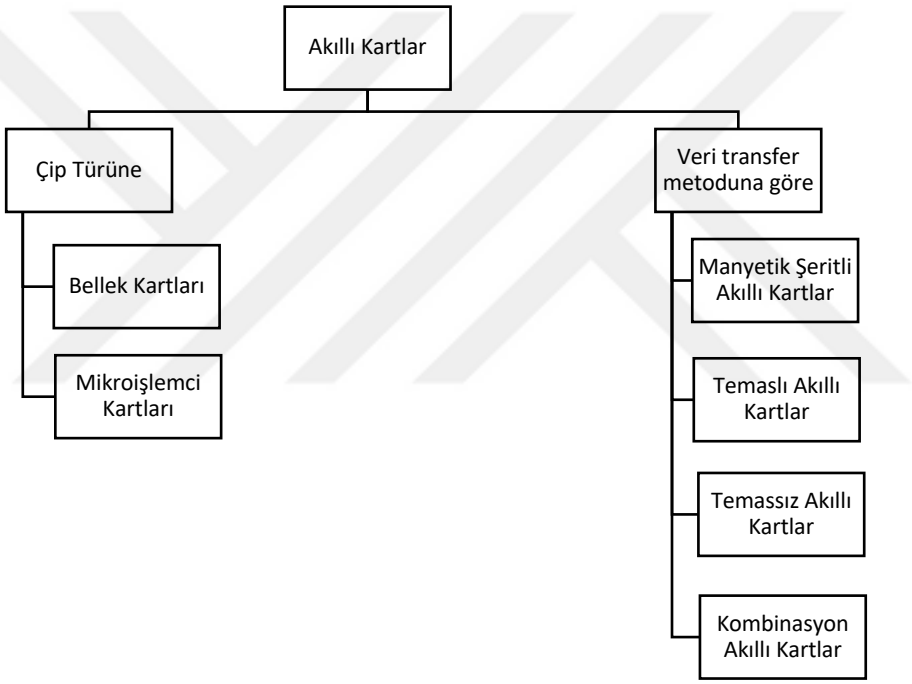
Banka kartlarında akıllı kart kullanılması telefon kartlarında kullanılması kadar kolay ve hızlı olmadı. Banka kartları için yarı iletken teknolojisinin gelişmesi yeterli değildi. Bunun yanı sıra modern kriptografinin de gelişmesi büyük bir önem taşımaktaydı. Elektronik veri işlemedeki gelişmeler sayesinde 1960 yılında kriptografide de önemli bir gelişme sağladı. Gelişen modern donanım ve yazılımlar karmaşık matematiksel algoritmaların uygulanmasına olanak sağladı. Bu sayede akıllı kartların güvenirliği geliştirildi. Böylelikle akıllı kartlar gizli anahtarı güvenli bir şekilde depolayabildiği ve kriptografik algoritmaları çalıştırdığı için üst düzey bir güvenlik sistemine sahip oldu. Böylelikle kullanımı için herhangi bir engel kalmamıştı. Çünkü manyetik şeritli kartlarda artan güvenlik zafiyeti ve riski yeni bir güvenlik sistemi barındıran banka kartlarına geçişi kolaylaştırdı. 1982-83 yıllarında Fransız bankaların yaptığı akıllı banka kartları denemesinden başarılı olmasından sonra 1984 yılında bunu uygulayan ilk ülke oldu. 1984-85 yılında ise Almanya ilk deneme olarak içinde yonga bulunduran çok işlevli ödeme kartı kullanıldı. 1997 yılında ise tüm Alman bankaları yeni akıllı kartlarını çıkardı. 2001 yılında ise Avusturya Pos işlemlerine sahip akıllı kartlar, elektronik cüzdan ve isteğe bağlı katma değerli hizmetler sundu. Bu sayede Avusturya dünya çapında elektronik cüzdan kullanımına sahip ilk ülke oldu (Rankl and Effing, 2010).

Gelişen teknoloji ve kriptografik gelişmeler sayesinde günlük hayattaki yerini korumaya devam etmektedir. Akıllı kartlar dünya üzerinde farklı alanlarda kullanılmaya başlandı. Akıllı kartlar günlük yaşantımızda toplu taşımada da "elektronik bilet" olarak temassız kart olarak kullanımı sağlanmaktadır. Temassız kartlar kullanımı kolay ve hızlı işlem yaptığı için günlük hayatımızda birçok alanda kullanılmaktadır.

4.1 Akıllı Kart Türleri

Akıllı kartları yonga türü ve veri transfer metoduna göre 2 ayrı başlık altında inceleyebiliriz.

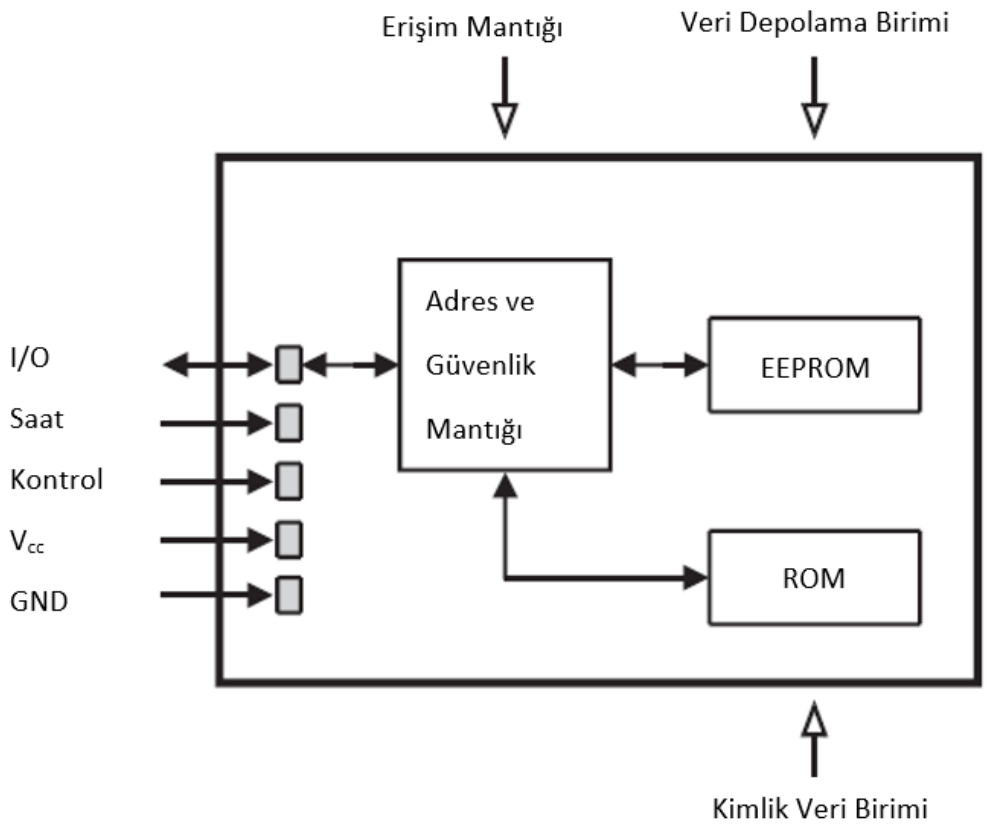
Tablo 1. Akıllı Kart Türleri.



4.1.1 Bellek Kartları

Bellek kartların kullanım amacı güvenli bir şekilde şifreleme için gizli anahtarı ve sertifikaları saklamaktır. Bellek içerisindeki gizli anahtarlar gerektiği durumlarda parola görevi görmektedir. Sertifikalar ise doğrulamayı sağlamaktadır (Shelfer and Procaccino, 2002). Bellekteki bilgiler EEPROM bellek kısmında

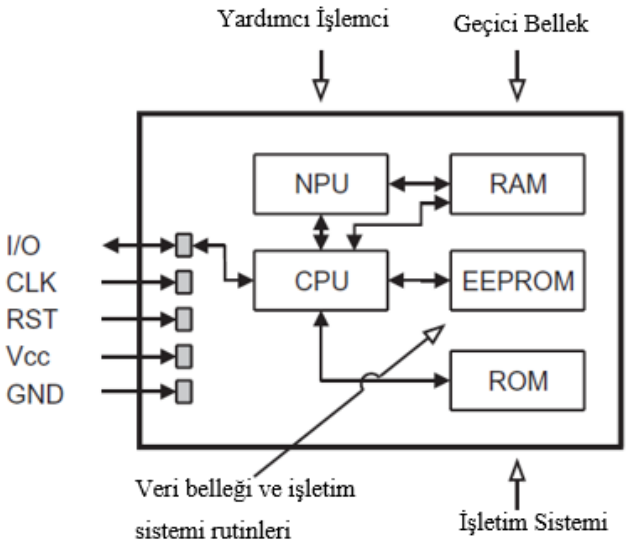
tutulmaktadır. EEPROM bellekler yazma koruması ve silme korumasından oluşan güvenlik mantığı tarafından yönetilmektedir. Veriler G/Ç (Giriş/Çıkış) bağlantı noktası sayesinde belleğe aktarılır ya da bellekten alınır. Bellek kartlarının kullanım alanı içerisindeki yonga ve işlevinden dolayı belirli alanlarda kullanılmaktadır. Genellikle telekomünikasyondaki telefon kartlarında, sağlık kartlarında kullanılmaktadır (Rankl and Effing, 2010). Şekil 14'te örnek bir bellek kartının tipik mimarisi verilmektedir.



Şekil 14. Bellek kartının tipik mimarisi.

4.1.2 Mikroişlemci kartları

Mikroişlemci kartlar bünyesinde (Şekil 15) veri saklama ve saklanan verilerin iletilmesi işlemi yanı sıra bu işlemlerin güvenlik protokollerinin gereksimini de sağlamaktadır. Kartın üzerinde ROM, EEPROM, RAM, işlemci VE G/Ç bağlantı noktasından oluşur. Rom yonga üretildiği zaman kart için yazılan işletim sistemini barındırır. EEPROM işletim sistemi tarafından gönderilen verileri ve programın yazıldığı yonganın kalıcı bellektir. RAM işlemcinin çalışma esnasında geçici verilerin bulundurduğu geçici bellektir. G/Ç ise mikroişlemci kartın dış ortama bit bit veri aktarımının sağlandığı yerdir. Mikroişlemci kartların en önemli öğesi işlemcidir. Yazılan işletim istemin çalışmasını sağlanması, kalıcı verilerin okunum programda çalışmasını ve verilerin güvenli bir şekilde dış ortama aktarılmasını sağlayan birimdir.

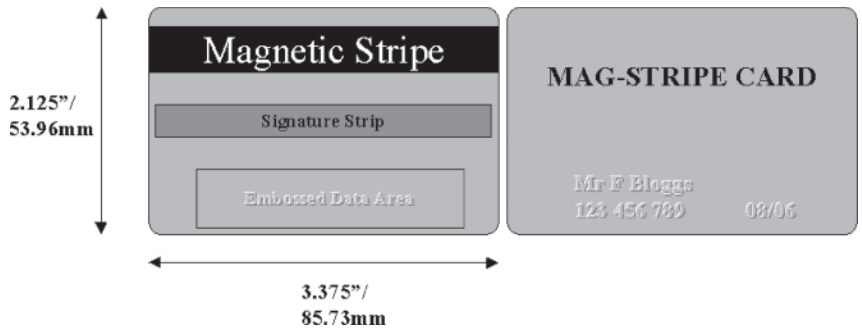


Şekil 15. Mikro işlemci kartların mimarisi.

Mikroişlemci kartlar, üzerlerine yüklenen yazılım sayesinde tek bir uygulama için kullanılmaktaydılar. Fakat gelişen teknoloji sayesinde üzerinde bulunan işletim sistemine yüklenen birkaç farklı uygulama sayesinde birden fazla alanda kullanımı sağlanmıştır. Bu gelişme sonucunda oluşabilecek güvenlik açıklıklarına engel olmak için özel donanım ve yazılım önlemleri kullanılmıştır. Bu işlemlerin gerçekleşmesi için yüksek işlem kapasitesi olan büyük bellek kapasiteli mikroişlemci yongaları da mevcuttur.

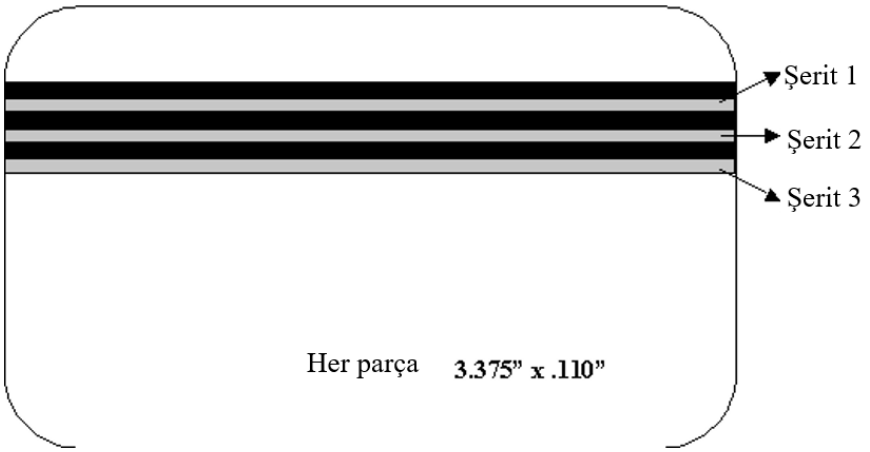
4.1.3 Manyetik şeritli kartlar

Manyetik şeritli kartlar (Şekil 16) üzerinde bulunan manyetik malzeme bandı içindeki demir bazlı manyetik parçacıkların manyetizması değiştirilerek içerisinde veri depolanmasını sağlar. Düşük maliyetli olmasından dolayı birçok alanda kullanılmaktadır. Akıllı kart tanımlamasının ilk 2 aşamasına uyan manyetik kartlar elektronik işlemlerde kullanılabilir, çoğu durumda güvenliği sağlamaktadır. Fakat 3.aşama olan kopyalanabilir ve taklit edilebilirliği olduğu için bazı durumlarda akıllı kart kategorisinde değerlendirilmemektedir. Böyle bir zafiyeti olmasına rağmen yıllarca kredi kartı ve banka kartı olarak kullanılmıştır (Mayes and Markantonakis, 2017). Günümüzde halen belirli alanlarda kullanılmaktadır.



Şekil 16. Manyetik kart örneği.

Manyetik şeritli kartların şerit kısmı Şekil 17'de gösterildiği gibi 3 parçadan oluşmaktadır. Şekil 16'da gösterildiği gibi Track 1 kısmı 79 alfabetik karakterden, Track 2 40 sayısal karakterden, Track 3 ise 107 sayısal karakterden oluşmaktadır. Bu üç parça içerisinde kartın sahibinin adı, adresi, CVV (Card Verification Value), PVV (Pin Verification Value), kartın son kullanma tarihi ile ilgili bilgileri bulundurmaktadır.

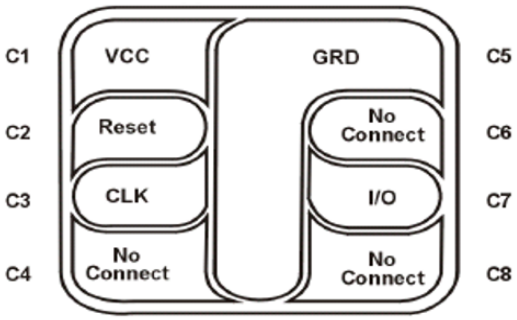


		KAYIT YOĞUNLUĞU	KAREKTER YAPISI	BİLGİ İÇERİĞİ
0,110" Şerit 1	IATA	210 BPI	7 bitlik karekter	73 Alfabetik Karekter
0,110" Şerit 2	ABA	75 BPI	5 bitlik karekter	40 Sayısal Karekter
0,110" Şerit 3	THRİFT	210 BPI	5 bitlik karekter	107 Sayısal Karekter

Şekil 17. Manyetik kartın mimarisi.

4.1.4 Temaslı akıllı kartlar

Akıllı karta verilen addan da anlaşılacağı gibi terminal ile temas gerektiren bir karttır. Temaslı kartlar plastik bir kartın üzerine yerleştirilen genellikle altın bazen de gümüş renkli yonganın olduğu karttır. Şekil 18'de temaslı kartın mimarisi gösterilmektedir. Yonganın üzerindeki sekiz bölmeden altı tanesi kart için etkin haldedir. Diğer iki bölme ise başka alanlarda kullanılabilmektedir (Rankl and Effing, 2010). Temaslı akıllı kartların da ISO-7816 standardında ki protokol kullanılmaktadır. Bu protokolde akıllı kartın fiziksel özelliklerini, temas yerini ve komunu, düşük ve yüksek seviyedeki uygulamaları, iletişim protokolü gibi özellikler belirtilmektedir (Dirk, 2001).



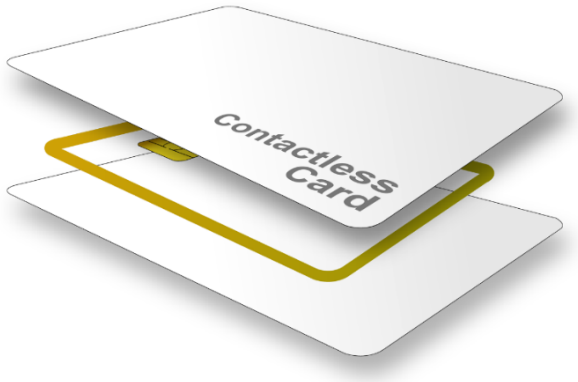
C1	VCC- Güç kaynağı girişi
C2	RESET- sıfırlama
C3	CLC- Zamanlama Sinyali
C4	Daha sonra kullanılmak üzere yedek alandır.
C5	GND- Topraklama
C6	VPP- Programlama voltaj girişi
C7	I/O- Çipe giriş/çıkış birimi
C8	Daha sonra kullanılmak üzere yedek alandır.

Şekil 18. Temaslı kartların mimarisi..

4.1.5 Temassız akıllı kartlar

Temassız kartlar herhangi bir temas olmadan işlemin gerçekleşmesini sağlayan bir karttır. İçerisinde bulunan pil sayesinde iletişim sağlamaktadır. Temaslı kartlar terminal ile temas halinde olduğu için kontaminasyon ve temas anında oluşan aşınmadan dolayı kart arızalanmaktadır. Temassız kartlardaki amaç bu sorunları ortadan kaldırmaktır. Ayrıca temassız kartların bir diğer avantajı ise zaman kazancıdır. Temaslı kartlarda işlemin yapılması için kartın terminal içerine takılması gerekirken temassız kartlarda bu işlem gerekmemektedir. Özellikle toplu taşımalarda temassız kart sayesinde işlem en kısa sürede gerçekleşmektedir.

Temassız kartlar (Şekil 19) içerisinde bulunan yonga herhangi bir işlem yapılmadığı durumda pasif haldedir. Yonganın aktif duruma geçmesi için terminal tarafından gönderilen manyetik alana girmesi gerekir. Temassız kart terminal tarafından gönderilen manyetik alana girdiğinde yonga aktif konuma geçerek radyo frekansları ile haberleşme sağlanır. Böylelikle işlem gerçekleştirilmiş olur.



Şekil 19. Temassız kart örneği.

4.1.6 Kombinasyon akıllı kartlar

Kombinasyon akıllı kartlarda ise hem temaslı hem de temassız özelliğini bir arada bulundurmaktadır. Böylece istenilen özellik kullanılarak işlem gerçekleştirilebilir. Temaslı kartı kullanmak için terminal ile temas sağlaması gerekir, temassız özellik için ise terminalin manyetik alanına girerek radyo frekansı ile işlem gerçekleştirebilmektedir.

4.2 Akıllı Kartlarda Kullanılan Şifreleme Yöntemleri

4.2.1 Veri şifreleme standardı

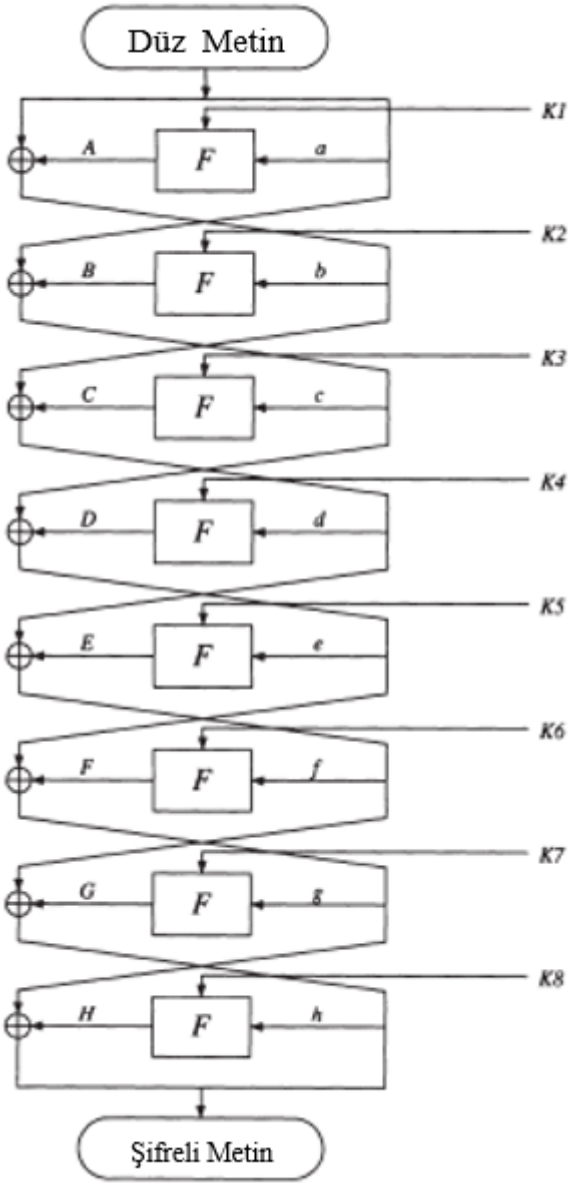
Simetrik algoritma türlerinden en çok kullanılan şifreleme sistemidir. DES şifreleme sistemi şifreleme ve şifre çözme işleminde aynı algoritma kullanılmaktadır. DES şifreleme sistemi 64 bitlik veri blokları 56 bitlik bir anahtar ile şifrelenir. Şifreleme işlemini gerçekleştirdikten sonra 64 bitlik veri permütasyon, ikameler ve XOR işlemlerini gerçekleştir (Standaert et al., 2006).

DES şifreleme sistemi ilk olarak ilk permütasyon işlemi uygulanır. Bu işlemin amacı bitlerin bir tablo yardımıyla yer değiştirmesidir. İlk permütasyon tablosu Tablo 2 de gösterilmiştir. Böylece 58.bit olan veri ilk permütasyon ile ilk biti, 7.bit olan veride sonuncu biti olmaktadır (Grabbe, 1992).

Tablo 2. İlk Permütasyon tablosu.

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

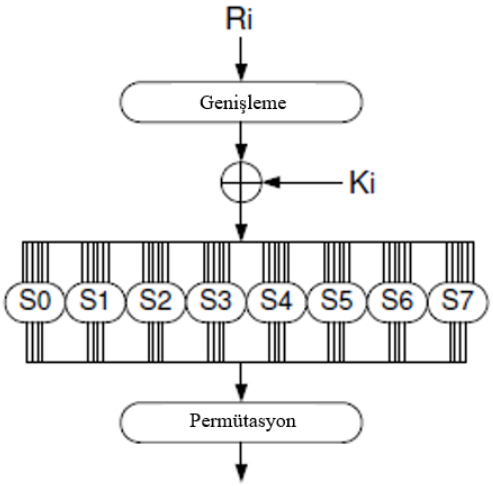
İlk permütasyon işleminden sonra G1 (sol) ve G2(sağ) olmak üzere 64 bitlik veri 2 eşit parçaya bölünür. G2 verisi ise 48 bitlik anahtar ile f fonksiyonuna girer. Buradan elde edilen 32 bitlik sonuç G1 verisi ile XOR işlemine girer. Bu işlemin sonucu 2.aşamanın yeni G2 verisi elde edilir. 2.aşamanın G1 verisi ise işlem görmemiş olan G2 verisi olur (Şekil 20) . Bu işlem 16 tur şeklinde uygulanır.



Şekil 20. DES algoritmasının tur örneği.

G2 verisine uygulanan *f* (Feistel) fonksiyonu Şekil 21'de gösterilmiştir. *F* fonksiyonu içerisinde bulunan *E* (Expansion) işleminde 32 bit olan veri

genişletilerek 48 bit yapılmaktadır. Daha sonra S-box işlemiyle tekrar 32 bitlik sonuç elde edilmektedir (Standaert et al., 2006).



Şekil 21. f fonksiyonu.

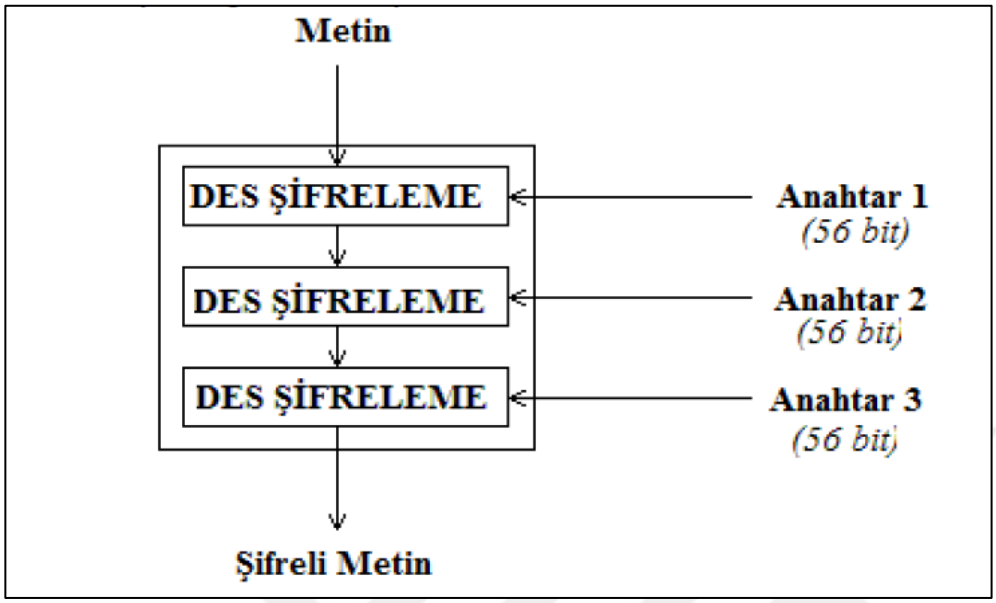
Bu aşamalardan sonra elde edilen 64 bitlik veri ters ilk permütasyona girer. Ters ilk permütasyon tablosu Tablo 3'te gösterilmiştir. Böylece 40.bit olan veri ilk permütasyon ile ilk biti, 25.bit olan veri de sonuncu bit olmaktadır (Grabbe, 1992).

Tablo 3. Ters ilk permütasyon tablosu.

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

4.2.2 3-DES

DES algoritmasının kusurlarının azaltılması için geliştirilmiştir. 3DES algoritmasında, DES algoritmasında uygulanan tüm aşamalar uygulanmaktadır. 3DES, algortimayı üç farklı anahtarla art arda üç kez uygulanmaktadır. Böylelikle 3DES algoritmasının anahtar uzunluğu 168 bittir (3x56bit). Şekil 22'de 3DES şifreleme örneği gösterilmiştir.



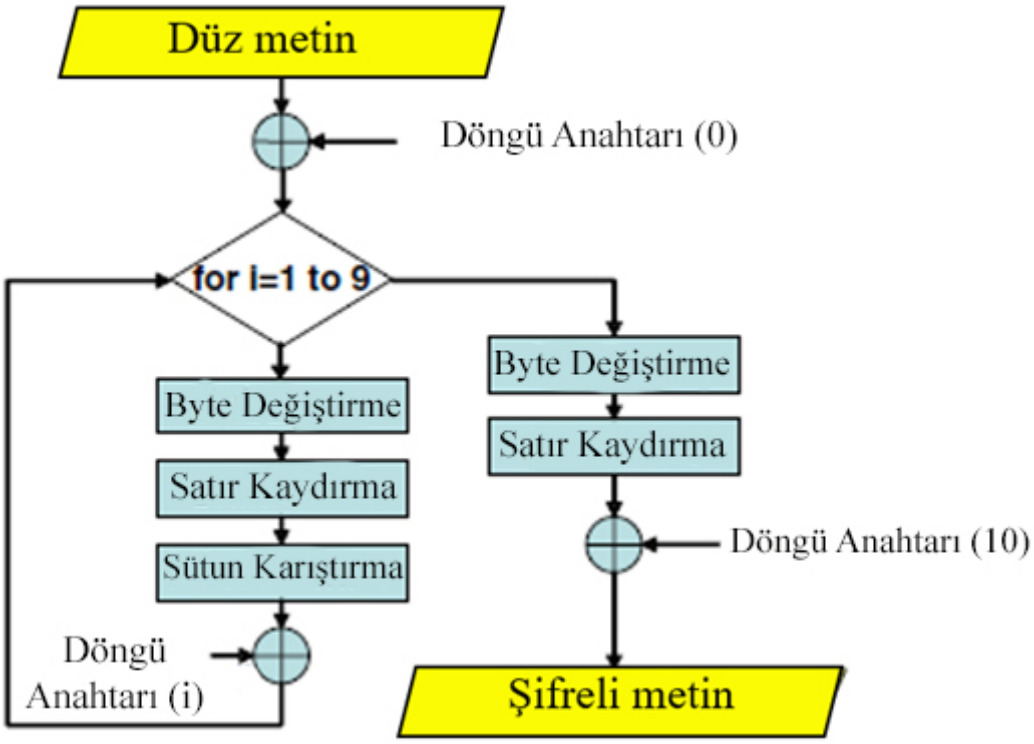
Şekil 22. 3DES şifreleme örneği.

DES şifreleme aşamasında DES algoritma kısmında yapılan tüm işlemler yapılmaktadır. 3DES'te DES'te bulunan ilk permütasyon (IP), f fonsksiyon işlemi ve ters permütasyon (IP^1) işlemleri bulunmaktadır (Aydoğan, 2016).

4.2.3 AES

Gelişmiş Şifreleme Standardı (AES,Rijndael), 2001 yılında ABD Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından oluşturulan elektronik verilerin şifrelenmesi ve şifrelenmiş verinin çözülmesinde kullanılan simetrik bir blok kodlayıcıdır. AES algoritmasının simetrik olmasının sebebi üretilen şifrenin hem şifrelemede hem de şifre çözmede kullanılmasıdır. AES algoritması her bir 128, 292, 256 bit boyutuna sahip blokları, 128, 192, 256 bit uzunluğunda anahtarlar ile şifrelemektedir. Standart şifrelemede blok ve anahtar uzunluğu 128 bit olan AES-128 kullanılmaktadır. AES şifrelemesi Bayt Değiştirme, Satır Kaydırma

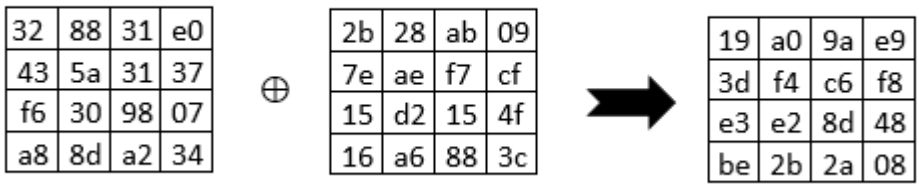
Sütun Karıştırma, Anahtar Toplama ve Anahtar Üretme işlemlerinden oluşur. AES için bu işlemler 10 kez tekrarlanır. Şekil 23'de gösterildiği gibi AES şifrelemesinde 9 tur boyunca kullanılan Sütun karıştırma işlemi 10. Turda yoktur (Di Natale et al., 2009).



Şekil 23. AES Algoritması.

4.2.3.1 Döngü anahtarı ve Bayt değiştirme (AddRoundkey and SubBytes)

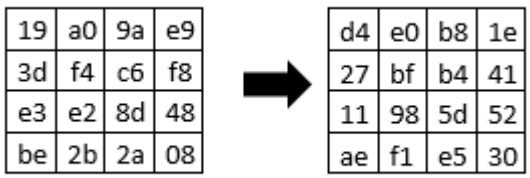
Şifrelenecek olan veri (düz metin) ve şifreleme anahtarının her biri 4x4 matrisler halinde düzenlenir. Bu matrislerde bulunan her bir değer 1 bayt veri tutmaktadır. Başlangıçta, düz metin başlangıç döngü anahtarı ile XOR işlemine girer. Bu aşamadan sonra bayt değiştirme (SubBytes) işlemine geçilir. Bayt değiştirme işlemi AES algortimasında doğrusal olmayan tek işlemdir. Döngü anahtarı (AddRoundkey) ile yapılan XOR işleminden elde edilen sonucun (Şekil 24) her değerinin karşılığı Rijindael S-box matrisinden aranarak o değerin yerine yazılır (Şekil 26). Rijindael S-box matrisi 16x16 lık bir matristir (şekil 25) ve AES algoritmasının değişim tablosu olarak da adlandırılmaktadır (Lo et al., 2017).



Şekil 24. Xor işleminden elde edilen matris.

	00	01	02	03	04	05	06	07	08	09	0a	0b	0c	0d	0e	0f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

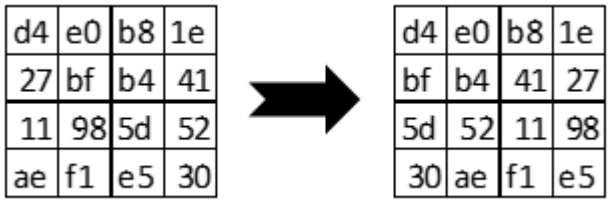
Şekil 25. Rijndael S-box.



Şekil 26. Byte değiştirme işlemi sonucu..

4.2.3.2 Satır Kaydırma

Satır kaydırma (Shift Row) işleminde bayt değişiminden sonra elde edilen 4x4 matris 1. satır aynı kalacak, 2. satır sağdan sola 1, 3. satır sağdan sola 2, 4. satır sağdan sola 3 birim kaydırılarak yapılan işlemdir. Bu işlem sonucunda elde ettiğimiz matris Şekil 27'de gösterilmiştir.



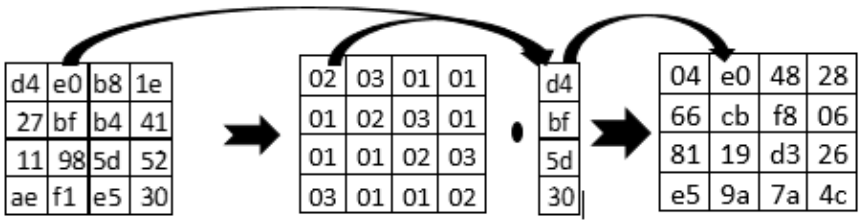
Şekil 27. Satır değiştirme işlemi sonucu.

4.2.3.3 Sütun Karıştırma

Sütun karıştırma (Mix Cloumns) işleminde her sütun ayrı olarak işleme alınır. Bu işleme alınan sütunlar 4 terimli polinom olarak kabul edilir. Bu sütunlar modulo $a(x) = x^4 + 1'de$

$$c(x) = 03 \cdot x^3 + 01 \cdot x^2 + 01 \cdot x^2 + 02 \qquad (4)$$

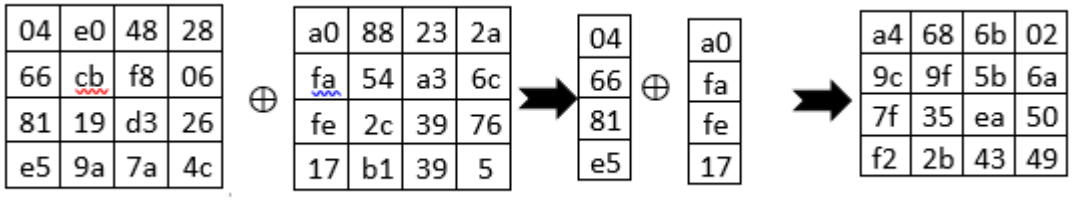
polinomuyla çarpılır. Sütun karıştırma işlemi AES algoritmasının son turunda yapılmamaktadır. Sütun karıştırma işleminin bir örneği Şekil 28'de gösterilmektedir.



Şekil 28. Sütün karıştırma sonucu.

4.2.3.4 Anahtar toplama

Her anahtar 16 bayttan oluşmaktadır. Bu anahtarlar sütun karıştırmadan elde edilen matris ile XOR işlemine girer (Şekil 29). Bu işlemden sonra 1.tur tamamlanmış oluyor. Bu aşamalar 9 tur devam etmektedir (Di Natale et al., 2009).



Şekil 29. Anahtar ile XOR işlemi sonucu.

4.2.3.5 Anahtar üretme işlemi

AES algoritmasında anahtar üretimi için bir önceki turun anahtarı işleme alınarak yeni bir anahtar elde edilir. Matris haline getirilen anahtara aşağıdaki

algoritma uygulanır (Şekil 30). Bu sayede bir sonraki turun anahtarı üretilir. Üretilen anahtar bir sonraki turun anahtar üretimi için kullanılır.

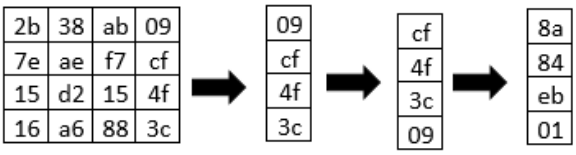
Require: N_k: Anahtardaki 32 bitlik kelime sayısı,
$w(i)$: Sütun numarası,
$RCon(i) \in GF(2^8)$,
$RCon(1) = x^0 = 01$, $RCon(2) = x^1 = 02$ ve
$RCon(i) = x \bullet (RCon(i-1))$

Ensure:

1: **for** i from 0 to $N_k - 1$ **do**
2: $w(i) = Key((32 \times i) - 1 : 0)$
3: **end for**
4: **for** i from N_k to $(N_b \times (N_r + 1)) - 1)$ **do**
5: **if** $i = 0 \bmod N_k$ **then**
6: $w(i) = w(i - N_k) \oplus (S - Box(Kaydır(w(i - 1)))) \oplus Rcon(i/N_k)$
7: **else**
8: $w(i) = w(i-1) \oplus w(i - N_k)$
9: **end if**
10: **end for**

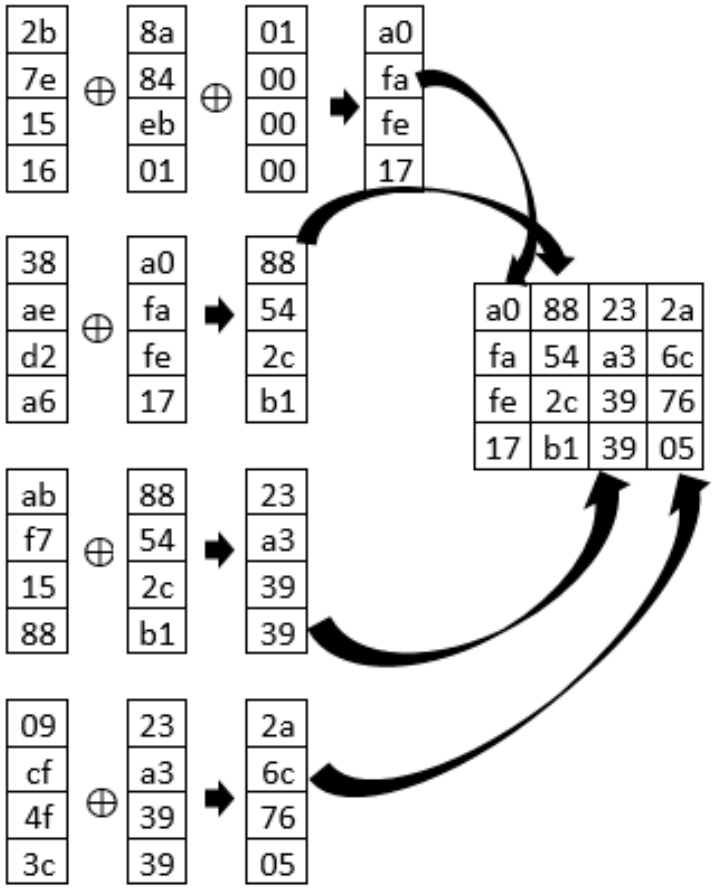
Şekil 30. Tur anahtar üretme algoritması.

Yukarıda belirtilen algoritma üzerinden yola çıkarak anahtarın aşamaları uygulayacağız.



Şekil 31. RotWord işlemi.

Bu aşamada başlangıç anahtarımızın son sütünü alınarak RotWord işlemi (Şekil 31) uygulanır. RotWord işleminde daha sonra her bir hücrede bulunan değerin S-box kutusundan değeri alınır. S-box'dan alınan değerler matris yerine yazılır (Şekil 32). Bu sütun 1 turun anahtarının ikinci ilk sütunun üretilmesinde kullanılır.

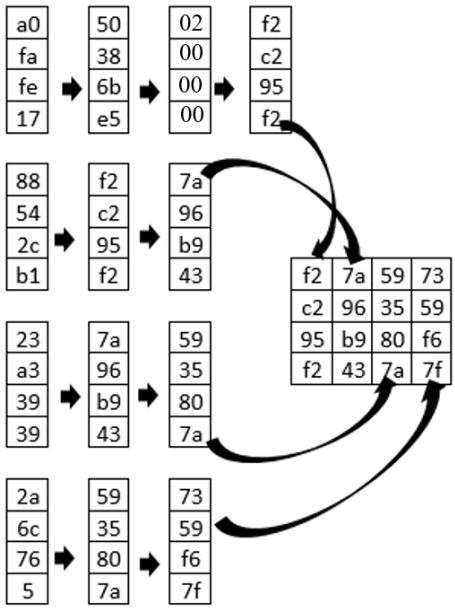


Şekil 32. AES birinci anahtar üretimi.

Elde edilen sütün ile birlikte ilk anahtarımızın birinci sütunu ve Rcon modülünün (Tablo 4) birinci tur sütunu XOR işlemine alınır. Elde edilen sonuç ikinci anahtarın birinci sütunun olur. Daha sonra birinci anahtarın ikinci sütunun ile elde edilen ikinci anahtarın birinci sütunu XOR işlemine girerek ikinci anahtarın ikinci sütunu elde edilir. Geriye kalan iki sütun içinde aynı işlemler yapılarak ikinci anahtar üretilir. Üçünücü anahtar üretiminde ise ikinci anahtardan yararlanılarak birinci anahtar üzerinde yapılan işlemler yapılarak elde edilir (Şekil 33).

Tablo 4. Rcon Modulü.

Tur 1	Tur 2	Tur 3	Tur 4	Tur 5	Tur 6	Tur 7	Tur 8	Tur 9	Tur 10
1	2	4	8	10	20	40	80	1b	36
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0



Şekil 33. AES ikinci anahtar üretimi.

Bu işlemler on tur içinde yapılarak AES şifrelemesi için gerekli olan tur anahtarları elde edilir (Rothke, 2007).

4.2.3.6 AES'in maskelenmesi

AES algoritmasına ait olan ara sonuçların maskelenerek gizlenmesi sağlayan bir uygulamadır. AES algoritmasına ait olan ara değerler *d*, maske olarak adlandırılan rastgele bir *m* değeri ile gizlenir. Maske değeri her değer için özeldir. Her çalışma için akıllı kartta yeni bir maske oluşturulur. Bu amaçla saldırı önlenmeye çalışılır. Maskelenmiş değer ara değer ve maske değerinin toplanması yani $dm = d \oplus m$ olur. Çarpımsal maskeleme de bulunmaktadır. Fakat akıllı kartlarda çarpan modülü için çarpımsal maskeleme uygulanamamaktadır.

Değerlerin maskelenmesi sonucunda kriptografik cihaz saldırganın tahmin edemeyeceği şekilde güç tüketimi yapmaktadır. Maskeleme işlemi algoritmanın ilk aşamasında eklenir. AES algoritmasının ShiftRows ve AddRoundkey aşamalarında herhangi bir enerji harcanmadan maskeleme gerçekleştirilebilir. MixCloumns aşamasında baytların karıştırılmasında belli bir çaba gerektirilirken SubBytes aşamasında ayrıntılı bir yaklaşım gerekmektedir. Tipik bir yazılım uygulamasında SubBytes işlemi tablo araması olarak uygulanır (S, SubBytes tablosunu belirtir). AES durumu 16 bayttan oluşur. Bu nedenle 16 tablo arama işlemi yapmamız gerekir. SubBytes işlemini maskelediğimizde, $S(dm) = S(d \oplus m) = S(d) \oplus m$ olacak şekilde maskelenmiş bir SubBytes tablosu S değerini hesaplamamız gerekir. Algoritmanın en sonunda, maskeler ara değerlerden çıkarılır (Herbst et al., 2006).

5. YAN KANAL ANALİZ UYGULAMALARI

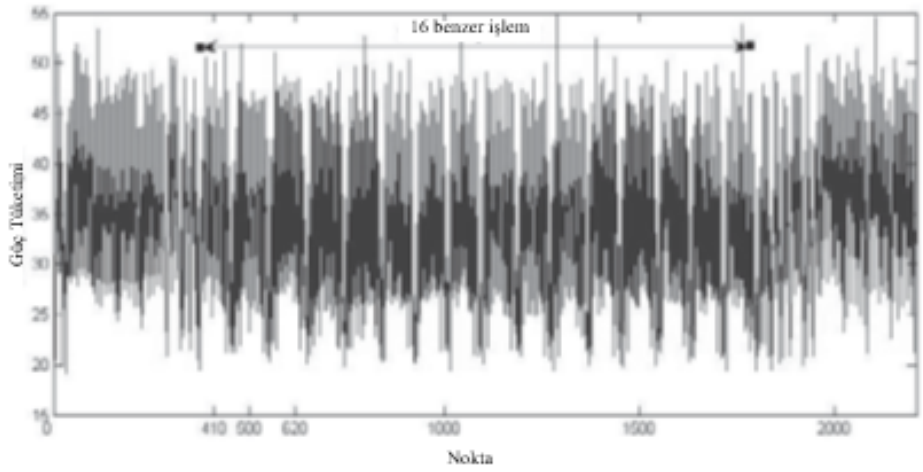
5.1 AES Algoritması Uygulamaları

Yapılan literatür çalışmalarında yan kanal saldırılarında AES algoritmasına yapılan saldırıların yöntemleri ele alınmıştır. Bu saldırılar farklı yöntemler kullanarak AES algoritmasının güvenirliği test etmektedir. AES algoritmasına yapılan saldırıları incelemek için iki farklı yöntem ele alınmıştır. Bunlar Oswald et al. ve Örs et al. yaptığı saldırılardır (Oswald et al., 2006; Örs et al., 2004).

Oswald et al. AES algoritmasının ilk adımı olan SubBytes adımın giriş ve çıkışına saldırı yapmayı planlamıştır. Saldırıyı gerçekleştirirken ikinci dereceden *DPA* saldırı türünü kullanmıştır. İkinci derece DPA saldırısında ilk aşama ön işleme adımı ve DPA adımından oluşmaktadır. Oswald et al. 8 bitlik micro cip bulunan maskeleme işlemi uygulanmış AES şifreleme sistemiyle güvenliği sağlanan akıllı karta saldırı gerçekleştirmiştir.

Burada öncelikle akıllı kart içerinde bulunan AES algoritmasının tüm aşamalarına maskeleme işlemi uygulamaktadır. Bu aşamada bayt değiştirme ve sütun karıştırmada maskeleme güvenlik için önemlidir. Çünkü bayt değiştirme aşamasında maskeleme işlemi ile S-box değişmektedir. Sütun karıştırmada ise 4 sütun için dört farklı maskeleme kullanılır. Buradaki amaç sütun karıştırma işleminde sütunun dört baytının birbiriyle birleştirilnesini sağlamaktır. Bu işlemlerden sonra ikincil dereceden DPA saldırısı ile akıllı kartta bulunan AES algoritmasının ilk şifreleme turunda iki S-box çıkışına saldırılmıştır.

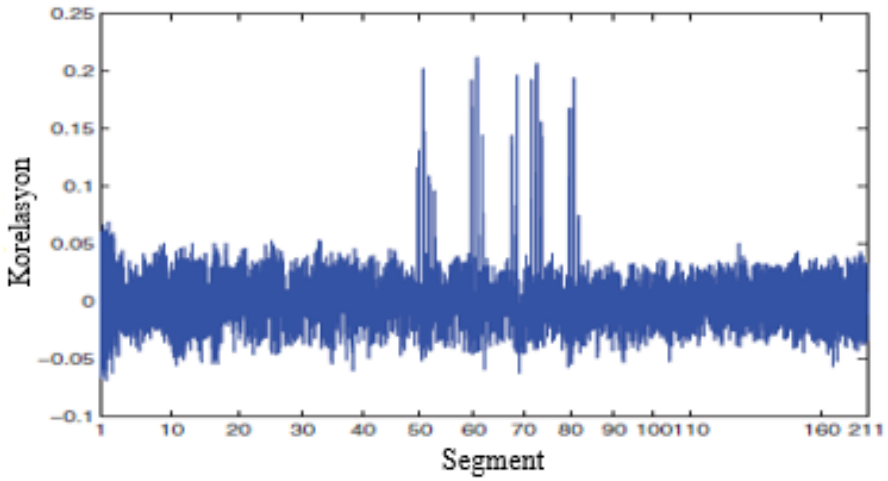
İlk şifreleme turunda iki S-box çıkışına yapılan saldırıda öncelikle bayt değiştirme işleminin özel veya iki çıktısının Hamming ağırlığı hesaplanmıştır. Bu hesaplama sonucunda tüketilen güç tahmin edimiştir. Daha sonra AES algoritmasının güç tüketimi ölçülerek güç izleri incelenmiştir. Bu izler sonucnda AES algoritmasının ilk turunun ne zaman gerçekleştiği ortaya çıkarılmıştır. Aşağıdaki Şekil 32'de gösterildiği gibi 16 benzer işlem tespit edilmiştir. Bu işlemlerin AES algoritmasının döngü anahtarı ve bayt değiştirme aşamalarına karşılık geldiği belirtilmiştir. Bu saldırıdaki amacın iki bayt değiştime işleminin çıktılarını hedef aldığı için Şekil 34'te görüldüğü gibi benzer işlemlerin gerçekleştidiği 410-620 aralığı seçilmiştir.



Şekil 34. İlk şifreleme turunun bir parçasının güç izinde 16 benzer işlem.

Bu işlemlerden sonra rastgele gerçekleştirlen AES algoritması sonucunda mikro denetleyicinin (kriptografik cihaz) güç tüketiminin 3000 adet ölçümü alınmıştır. Daha sonra ikincil DPA ön işlemi uygulanmıştır. Bu ön işlemde 410-610 aralığındaki tüm noktaların mutlak değeri hesaplanmıştır. Yapılan heaplamalar

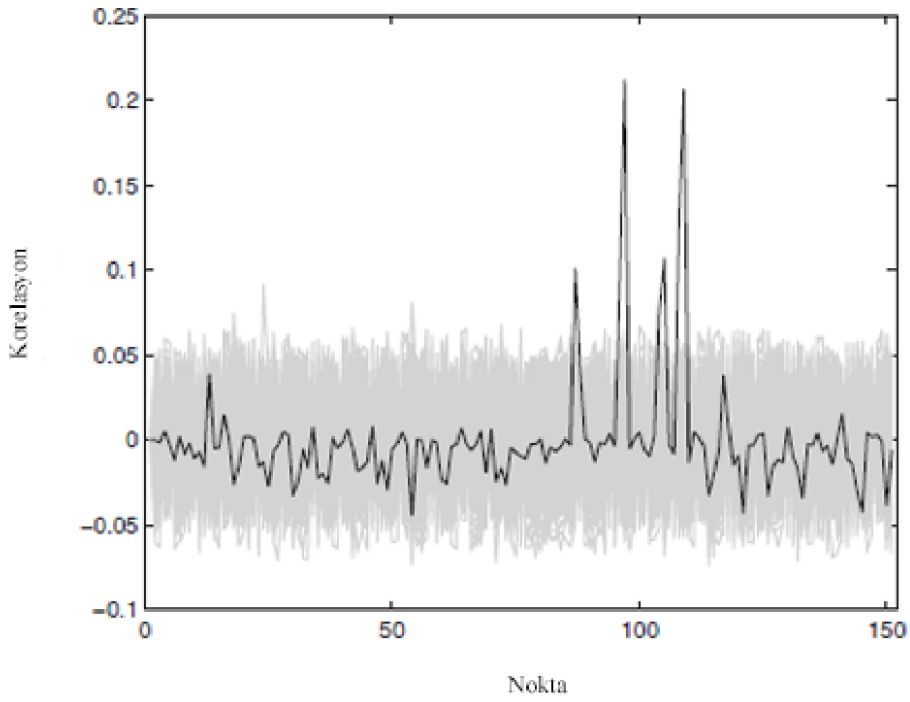
sonucunda her bir güç izine ait 210 parça elde edilmiş ve her güç izine ait olan 210 parça birleştirilmiştir. Bu izlere dayalı olarak Hamming ağırlık değeri (HW(S(P1 ⊕ K1) ⊕ S(P2 ⊕ K2))) için oluşturulan değerlere standart bir DPA saldırısı gerçekleştirilmiştir. Ara sonuç Hamming ağırlık değeri iki anahtar bayta bağlı olduğu için 65536 anahtar tahmini gerek duyulmuştur. Doğru anahtardan elde edilen sonuç Şekil 35'te gösterilmiştir. Bu sonuçta birden fazla tepe olmasının sebebi mikrodenetleyicinin çok kez manipüle edilmesinden kaynaklanmıştır. Şekil 3'de görüldüğü gibi en yüksek tepe segment 61 de elde edilmiştir. Bu segment orjinal izlenimlerde 471 ile 620 ve 470 noktaları arasına yapılan ikincil dereceden DPA saldrısının sonucunu içermektedir (Şekil 35). Bu segment 150 değerden oluşur.



Şekil 35. Orjinal güç izlerinin 410 ila 620 aralığında ikinci dereceden bir DPA saldırısının sonucu.

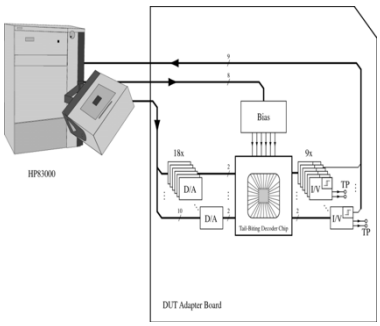
İkinci dereceden DPA saldırısında yalnızca doğru anahtarın bir tepe ürettiğini görmek için 65536 anahtar hipotezinin tümüne dayanarak 150 noktaya saldırılmıştır. Bu saldırının sonucu Şekil 36'da gösterilmektedir. 65535 hatalı

tuşların sonuçları gri renkte, doğru anahtarın sonucu ise siyah renkte gösterilmiştir. Yalnızca doğru anahtarda beklenen tepe noktaları oluşmuştur (Oswald et al., 2006).

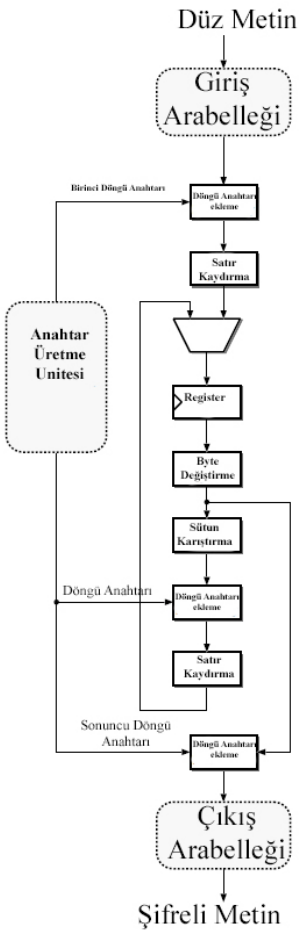


Şekil 36. Yapılan bir saldırıda tüm 65536 anahtar tahminin sonucu.

Örs et al. ölçümleri yapmak için HP83000 (Şekil 37) test sistemi ve Tektronix 784c örnekleme osiloskobu ve Fastcore uygulamasını kullanmıştır. Fastcore AES algoritmasının verimli olarak çalıştıran bir uygulamadır. Şekil 38'de Fastcore'un şifreleme veri yolu yapısını gösteren blok diyagram gösterilmiştir. Bu devrede sadece çekirdeğe ait güç tüketimi ölçülmüştür. Anahtarlanma gürültüsünün azaltılması için ölçüm 16 defa tekrarlanmıştır.



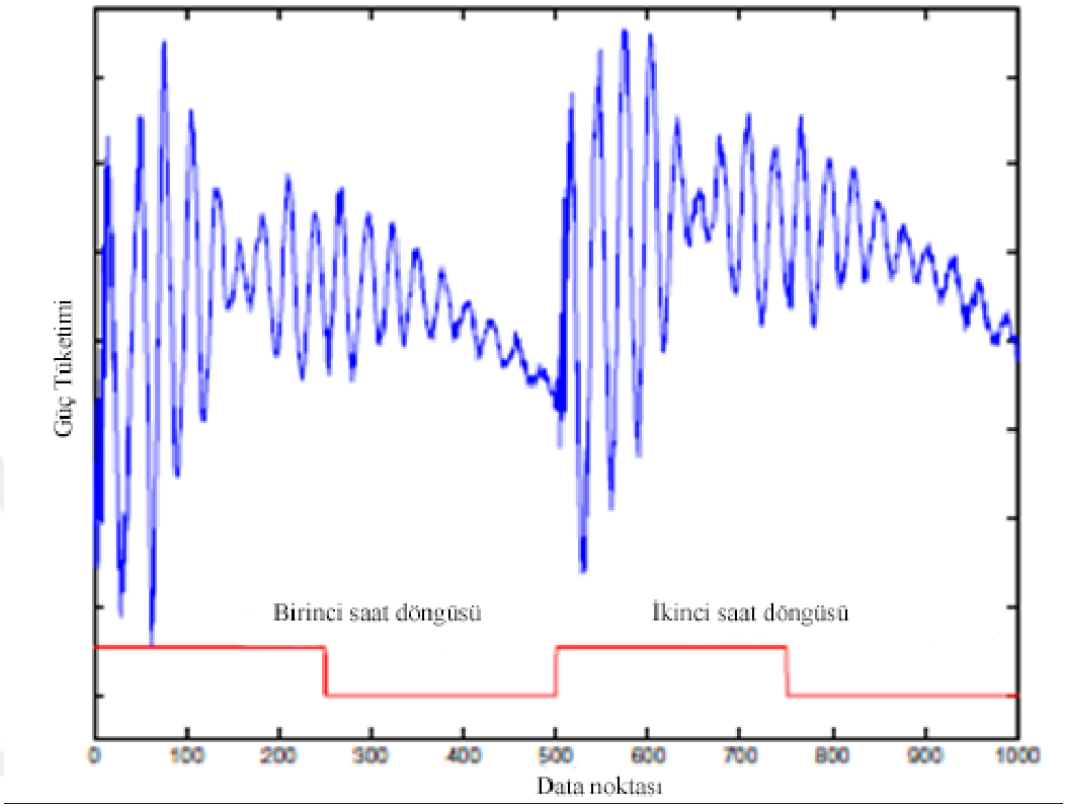
Şekil 37. HP83000 test sistemi.



Şekil 38. Fastcore kripto yongasının şifreleme yolunu vurgulayan blok diyagram.

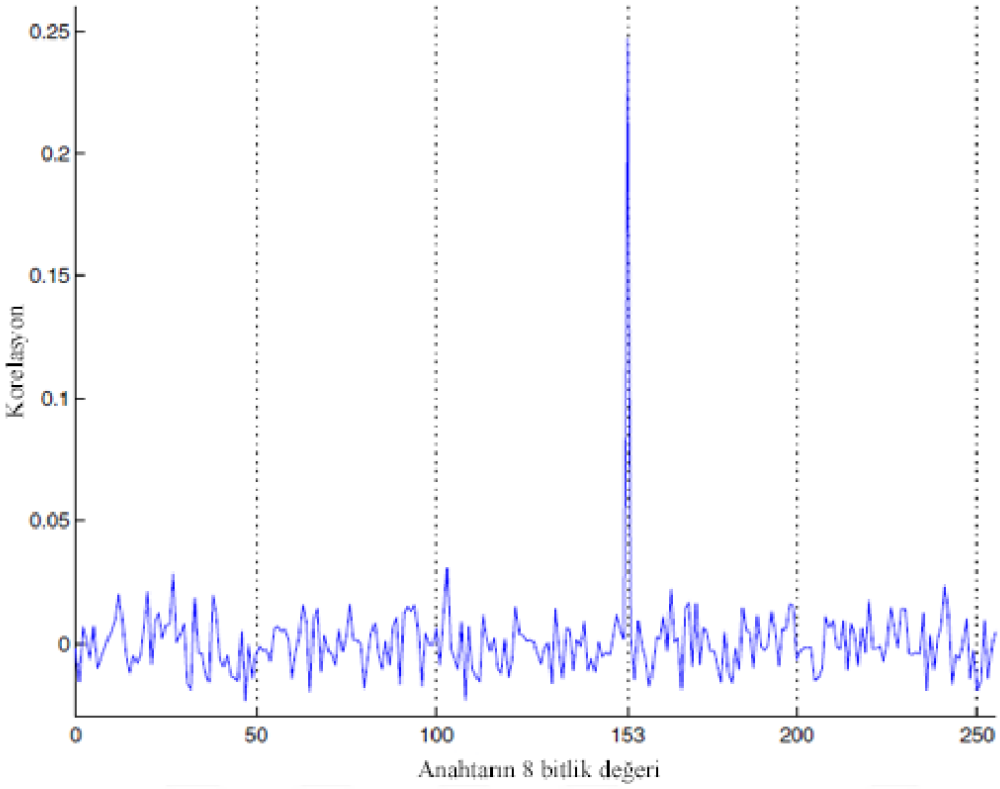
DPA saldırısının hedefi, ilk anahtar işleminden sonra registerdaki 8 bit hedef alınmıştır. Çünkü register aşamasına kadar 8 bitin konumunda herhangi bir değişim olmadığı için bu bitlerin depolanması sırasındaki güç tüketimi ölçülmesi planlanmıştır. Bu planlanmalar Örs et al. tarafından öncelikle simülasyonda denenmiştir. Daha sonra bu simülatorden elde edilen bilgiler doğrultusunda bir DPA saldırısı gerçekleştirilmiştir.

Simülator denemesinden kullanılan N düz metni anahtar ile şifreleme yapılmıştır. İlk anahtar işlemi algortimanın ilk turunda gerçekleşmektedir. Bu işlem sonucunda elde edilen değerler ikinci turda Register'a kaydedilir. Bu nedenle şifreleme işlemi yapılırken algoritmanın ilk iki turunda Register'ın güç tüketimi ölçülmüştür. Yongaya uygulanan tur frekansı 2 MHz ve osiloskop örenekleme frenaksı 1 GHz olarak ayarlanmıştır. Bu ölçüm değerleri sonucunda tur başına 500 örnek alınmıştır. Bu ölçümler sonucunda Nx100 matris (N=10000) değerinde bir M5 dosyası elde edilmiştir. Bu ölçümlerde elde edilen güç izi örneği Şekil 39'da gösterilmiştir.



Şekil 39. 50. güç izine ait ölçüm.

Örs et al. simülasyonda L=8 bit olacak şekilde bir DPA saldırsı deneyi gerçekleştirmişir. Bunun sonucunda $Nx2^L$ matris M4 hesaplamışlardır. Buradaki M4 matrisi ilk anahtar eklendiğinde L saldırılı anahtar bitlerinin belirli bir tahmin için bit değişikliklerini barındırmaktadır. Bu aşamalar sonucunda L anahtar bitlerine karşılık gelen yalnızca bir değerin yüksek bir korelasyona yol açtığı Şekil 40'ta görülmüştür.



Şekil 40. L bitlerinin korelasyon grafiği.

Bu işlemin gerçek bir DPA saldırısında anahtarın doğru L bitlerini belirlemek için korelasyon katsayısı kullanılmıştır. Ayrıca elde edilen ölçümlerde gürültünün ve veri miktarının fazla olmasından dolayı bir ön işlem uygulanmıştır. Ön işlem tekniği turların ortlamasından oluşmaktadır. Birinci ve ikinci tura ait tur ölçümlerinin ortalaması hesaplanmıştır.

$$e_{i,j} = E(M5(i, D * (j-1) + 1D * j)), \qquad (5)$$

i=0,…,N kadar düz metin

j=1. ve 2. tur

D= Bir turda ölçülen veri noktalarının sayısı

Ayrıca ilk turda akımın DC bileşeninden kaynaklanan gürültün oluştuğu için ikinci turun ortalama değerlerini çıkararak gürüntünün bazı kısımları kaldırılmıştır. Bu işlem sonucunda M6 matrisi oluşmuştur.

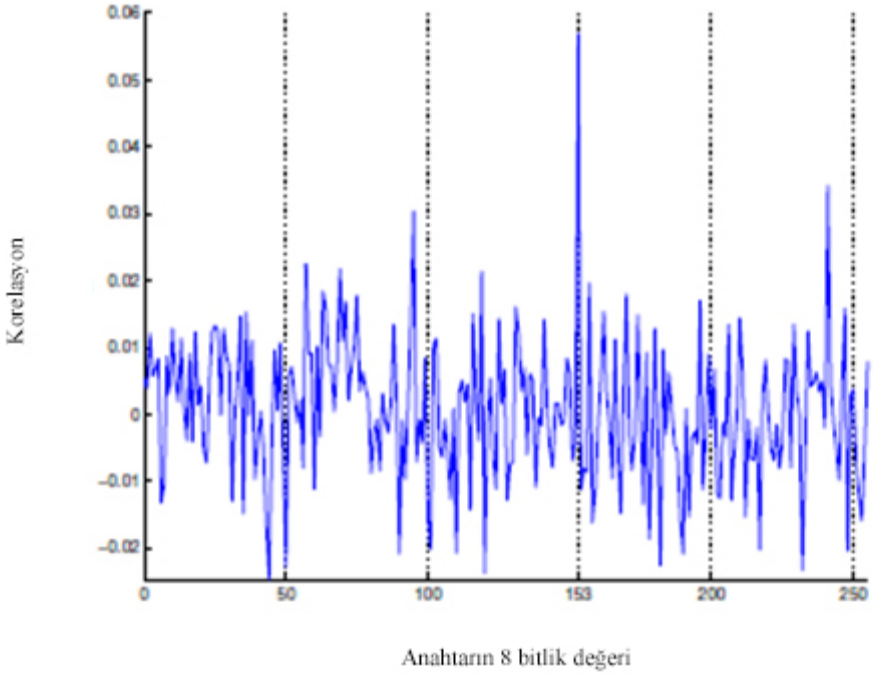
$$M6(i) = e_{i,2} - e_{i,1}, \quad (6)$$

Bu ölçümler korelasyon analizi için girdi olarak kullanılmıştır.

$$c_i = C(M6, M4(1{:}N, i)),$$

i=0,…,2L-1

Korelasyon analizi sonucunda en yüksek korelasyon i=153'te meydana gelmiştir (Şekil 41). Bu değer 8 bitlik anahtarın 0x99 bite karşılık gelmektedir.



Şekil 41. M4 ve M6 sütunları arasındaki korelasyon grafiği.

Fastcore bir turu yaklaşık 7ns de tamalamaktadır. Bu yüzden ölçümün ilk veri noktaları saldırı işlemi ile ilgili bilgileri barındırmaktadır. Bu yüzden ön işlem adımı için veri noktalarının miktarı azaltılmıştır. Bunun için önceden işlenmiş ölçüm verileri arasındaki korelasyon katsayısı hasaplanmıştır. M4'ün doğru anahtarına karşılık gele sütun;

$$M7(i,j) = E\big(M5(i, D+1{:}D+j)\big) - E\big(M5(i, D+1-j{:}D)\big),$$

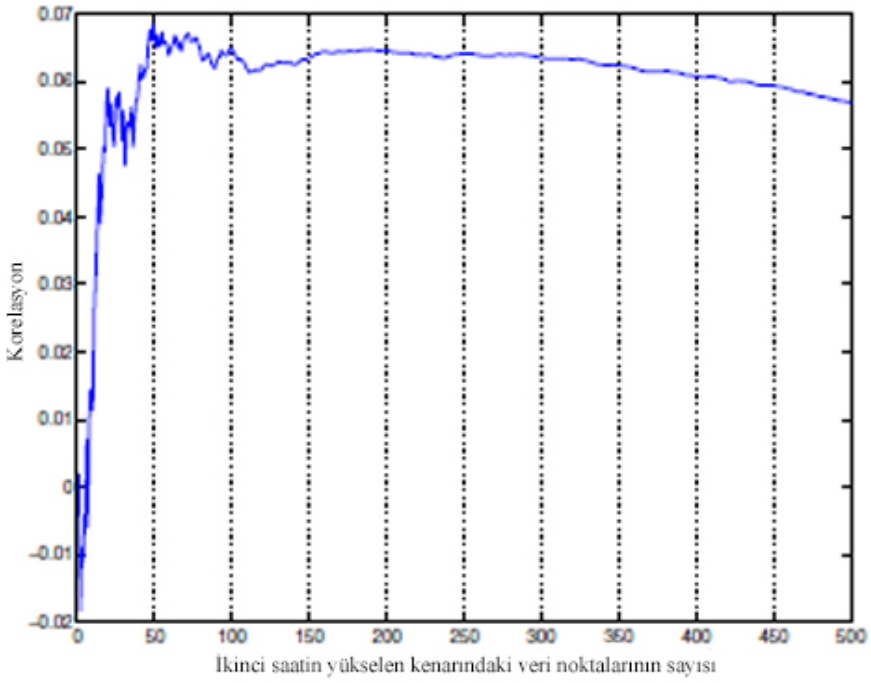
$$i=1,\ldots,N \qquad (7)$$

$$j=1,\ldots,D$$

$$c_i = C(M7(1{:}N, i), M4(1{:}N, 153)),$$

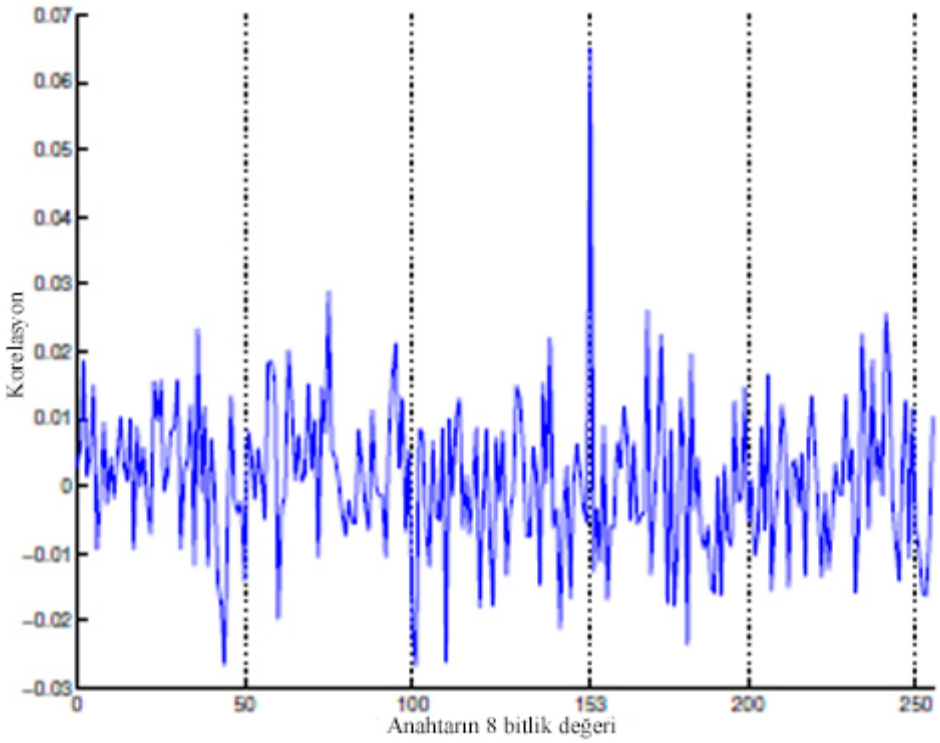
$$i=1,\ldots,D$$

İkinci turda 50 veri noktası kullanıldığında korelasyonun en yüksek olduğu Şekil 42'de gösterilmiştir.



Şekil 42. 50 verinin korelasyonu.

Şekil 43'te ise M4'ün tüm sütunları ile M7'nin 50 sütunundaki önceden işlenmiş veriler arasındaki korelasyon katsayıları gösterilmiştir. Bu şekilde elde edilen tepe noktasının doğru anahtara karşılık geldiği, yanlış anahtara karşılık gelen tepelerin sabit kaldığı gösterilmiştir.



Şekil 43. M4'ün tüm sütunları ile M7'nin 50. sütunu arasındaki korelasyon.

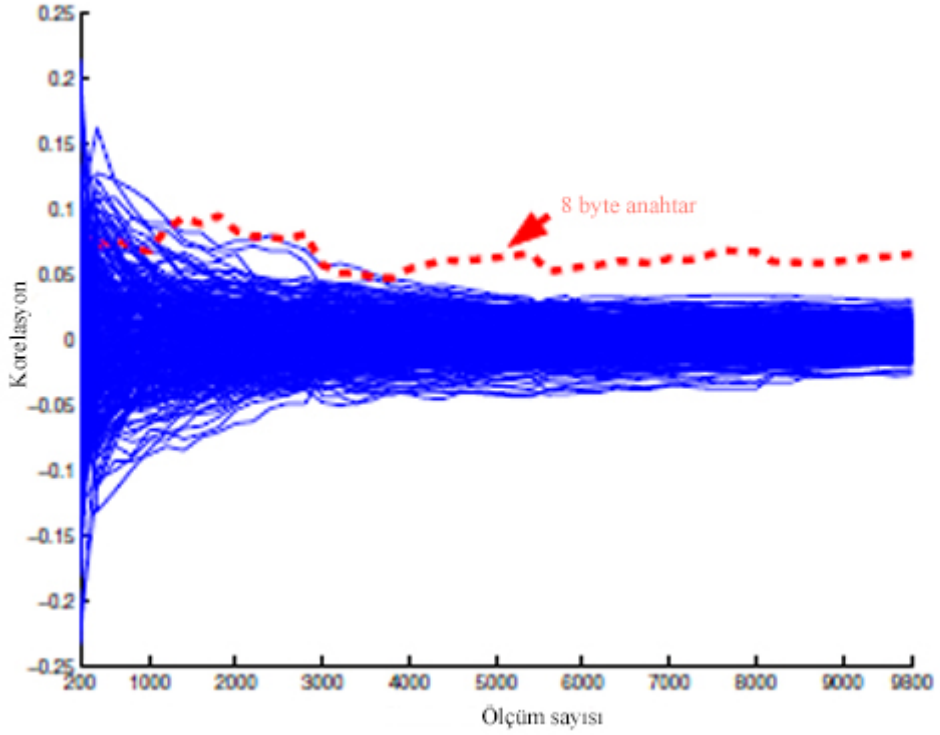
Örs et al. ekibi doğru anahtara ulaşabilmek için minimum sayıda ölçüm için korelasyon katsayılarını hesaplamışlardır.

$$c_{i,j} = C(M7(1{:}\,i, 50), M4(1{:}\,i, j)),$$

$$i=1,\ldots,N \qquad (8)$$

$$j=0,\ldots., 2^{L}\text{-}1$$

Bu hesaplamalar sonucunda Şekil 44'te gösterildiği gibi 4000 ölçümden sonra anahtarın doğru ve yanlış 8 bitin ayırt edilebileceği gösterilmiştir (Örs et al., 2005).



Şekil 44. Farklı ölçüm sayıları için M4'ün tüm sütunları ile M7'nin 50. sütunu arasındaki korelasyon.

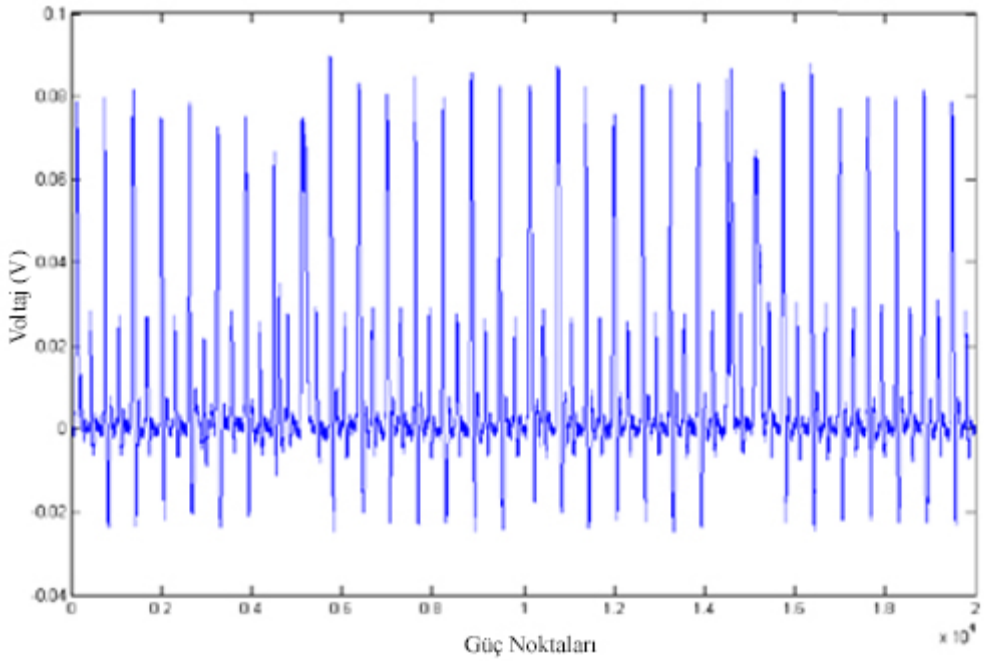
5.2 DES Algoritması Uygulamaları

Yapılan literatür çalışmalarında yan kanal saldırılarında DES algoritmasına yapılan saldırıların yöntemleri ele alınmıştır. Bu çalışmalar farklı şekilde yöntemler uygulayarak DES algoritmasının güvenirliği test edilmektedir.

DPA saldırısı DES şifrelemesinin iki özelliğine dayanır. Birincisi DES şifrelemesinde S-box çıkışlarının hedef cihazdan kaydedilen güç bilgileriyle bağlantılı olack şekilde hassas veriler üretmesidir. İkinci özellik ise her S-box girişinde o tur için kullanılan 48 bitlik alt anahtarın yalnızca 6 bitini kullanmasıdır. DES şifrelemesine yapılacak olan saldırının amacı şifrelemeya ait olan ilk turun alt anahtarının bulmaktır. Bu anahtarın şifreyi çözmeye yetecek kadar olan kısmı belirlendikten sonra, DES şifrelemesinde kullanılan tam anahtara ulaşılabilmektedir. Hnath ve Pettengil bu saldırıyı gerçekleştirmek için MATLAB programını kullanmışlardır. (Hnath and Pettengill, 2010)

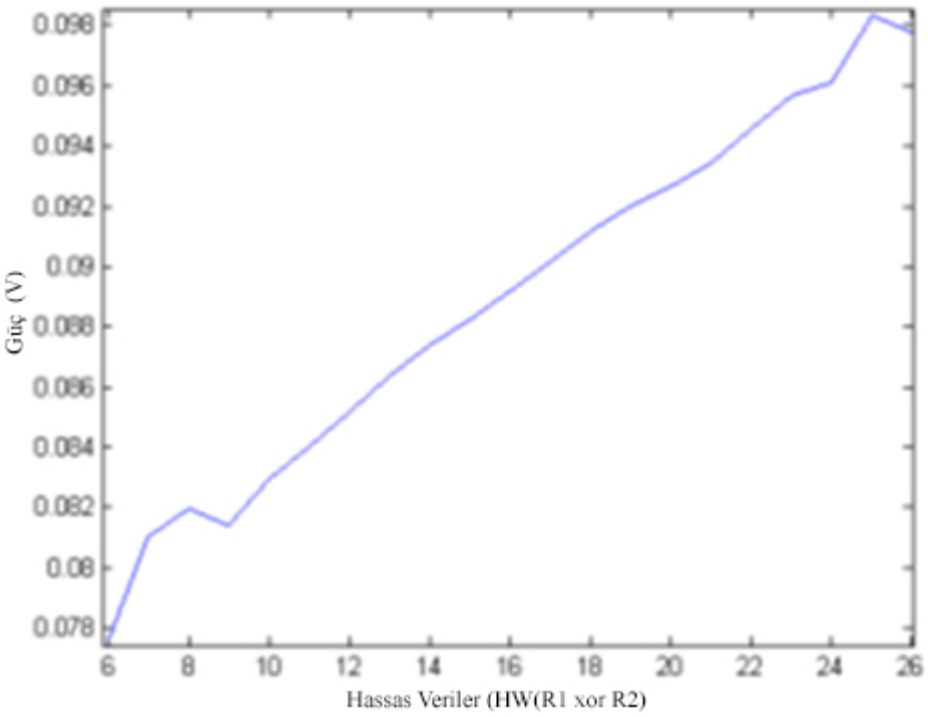
MATLAB'ta hazırlanan DPA saldırı programının ilk adımı güç bilgisi ile ilişkili hassas verilerin belirlenmesidir. Bunun için DES şifrelemesinin ilk turunda girdi ve çıktılara bakılmıştır. DES şifrelemesinde LR'nin 64 bitlik düz metni barındığı bilinmektedir. Bu düz metnin sağ 32 biti R1, DES şifrelemesininin Feistel işlevine girilen bitleri içermektedir. DES şifrelemesinin ilk turu tamamlandıktan sonra LR yeni bir değerle güncellenmektedir. Yeni LR değerinin sağ 32 biti R2 anahtarla bağlantılı olduğu için güç saldırısı buraya gerçekleştirilmektedir. Güç analizinin yapılması için gereken R1 ve R2 bitleri arasındaki Hamming mesafesi program için hassas veri olaarak kabul edilmektedir.

Bu bilgiler doğrultusunda program güç bilgisi için güç izleri almaktadır. Her DES güç izi 20x109 örnek/s hızında alınan 20000 nokta güç bilgisinden oluşmaktadır. $P_i = (p_i(t_1), p_i(t_2), \dots, p_i(t_m))$ t_i zamanında devreye ait voltajı gösteren nokta güç bilgilerinin toplamı olarak kabul edilmektedir. Alınan güç izlerine bakılarak DES şifrelemesinin 16 turuna ait 16 adet güç tepesi tanımlanmaktadır (Şekil 45).



Şekil 45. Şifreleme sırasında bir DES devresinin güç tüketiminin yan kanal okumasından ölçülen sonuçlar.

Birinci tura ait olan maksimum noktaya ait güç bilgisi p_{max} olarak kabul edilmiştir. Hnath ve Pettengill 4000 güç izi incelendikten sonra her ize ait olan ilk tur güç tüketimi tepe noktası (p_{max}) ve ilk tura ait olan hassas veriler ($HW(R_1 \oplus R_2)$ alınmıştır. Her hassas veri için 0'dan 32 bite kadar, o hassas veri değerini içeren tüm güç izleri boyunca güç tüketimi tepe değerlerinin ortalaması alınmıştır (Şekil 46).

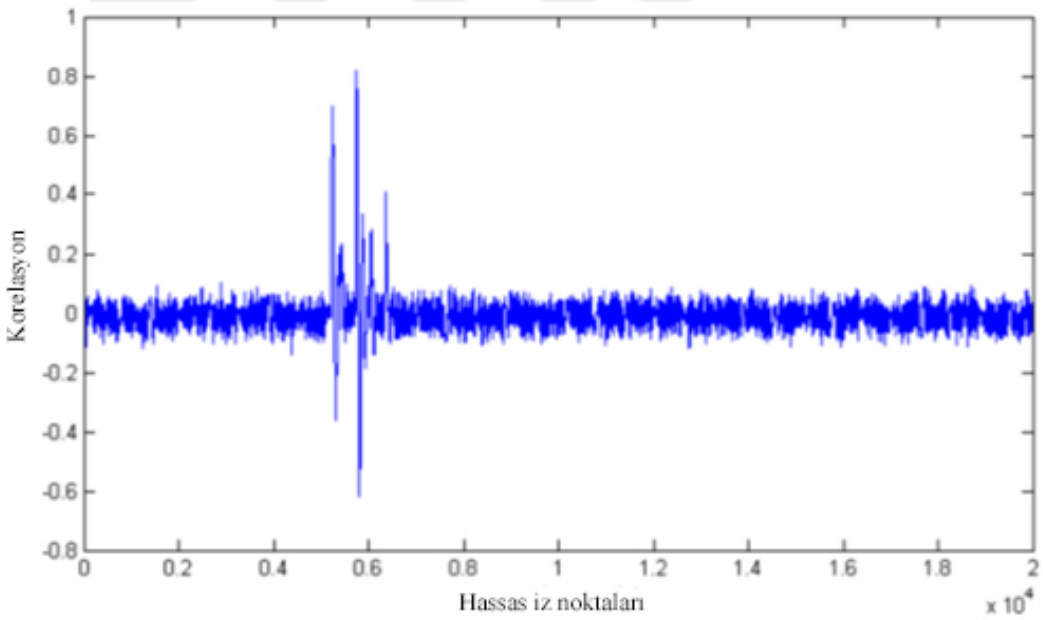


Şekil 46. Hamming mesafesine karşı güç tüketimi.

Elde edilen sonuçların hassas veriler ile DES şifrelemesinin ilk turuna ait güç tüketimi arasında bir ilişki olduğu gösterilmiştir. Hamming mesafesini ile devrenin güç tüketimi arasında doğru orantı vardır. Bu yüzden güç tüketimi değerleri ile Hamming mesafesi değerleri arasında örnek bir korelasyon belirlemek mümkündür. Güç izi değerleri ile hassas veriler arsındaki korelasyon Pearson katsayı denklem 9 kullanılarak hesaplanmaktadır.

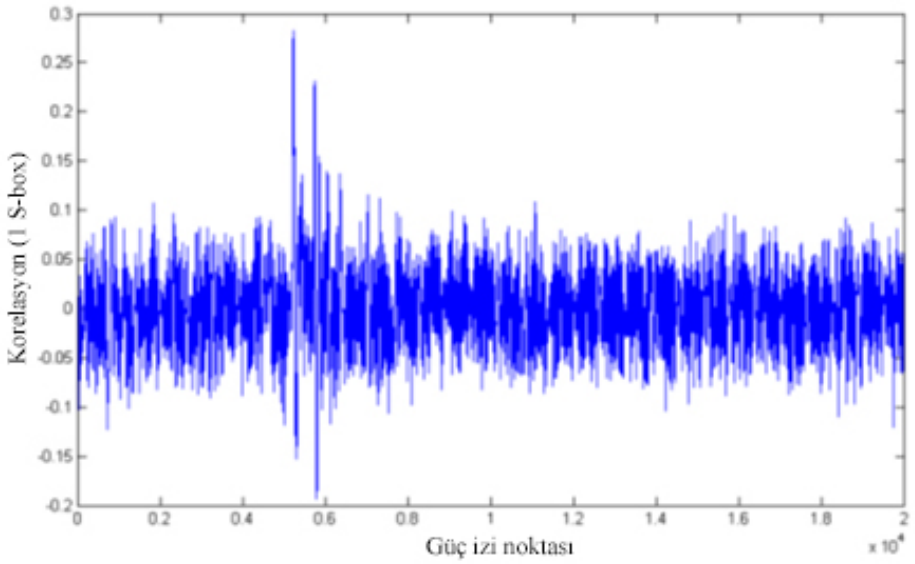
$$r_{sb} = \frac{\sum_{i=1}^{N}(h_i^{(sb)} - \overline{h^{(sb)}})(p_i(t_j) - \overline{p(t_j)})}{(n-1)\sigma_h \sigma_{p(t_j)}} \qquad (9)$$

Bir t_j noktasının tüm izleri boyunca ortalama değer $\overline{p(t_j)}$ olarak verilmektedir. Bu değerlerin bir t_j noktasındaki standart sapması $\sigma_{p(t_j)}$ olarak verilir. Her iz *i* için, yok sayılabilecek hassas veriler vardır. *i* izi için hassas veriler h_i, tüm izlemelerdeki ortalama hassas veriler $\overline{h_\iota}$ ve standart sapma σ_h ile birlikte hi olarak gösterilmektedir. Güç bilgisini ilk turdaki hassas verilerle ilişkilendirmek için bu denklemi kullanarak, ilk turda korelasyonda bir artış olması beklenmektedir. Bunu test etmek için, 1000 toplam güç izi için her bir güç izindeki her noktada (j = 1 ila 20000) bu katsayıyı hesaplanmaktadır. (daha fazla iz kullanıldıkça korelasyon artar, yüksek korelasyon değerleri üretmek için 1000 iz yeterlidir). Sonuçlar Şekil 47'de gösterilmektedir.



Şekil 47. Güç izlerindeki her j noktası için ilk tur hassas veriler ile güç izleri arasındaki korelasyon.

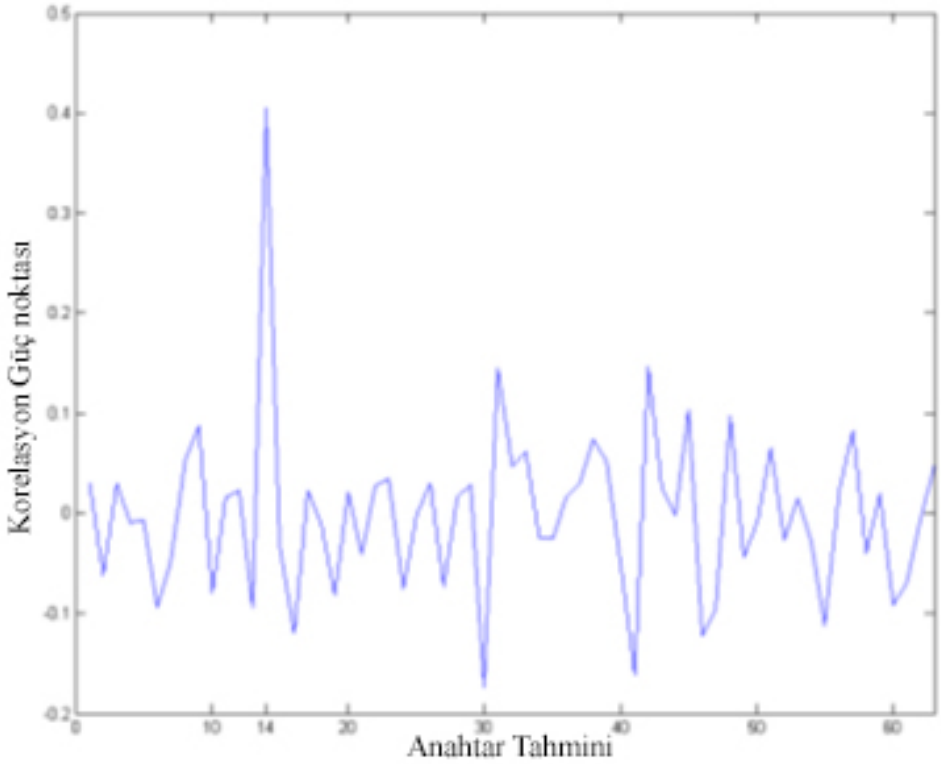
Bu grafikten, hassas veriler ile j = 5750 etrafında merkezlenen güç izi arasında güçlü bir korelasyon olduğu görülmektedir. Bu sayede Hamming mesafe modelinin DES üzerindeki etkisi daha da doğrulanmaktadır. Fakat saldırının tam mümkün olabilmesi için anahtar bitlerinin kurtarıldığından daha fazla emin olunması gerekmektedir. Bunun için yarım baytlık hassas veri (tek bir S-box çıkışına karşılık gelen) verildiğinde veriler arasındaki korelasyonun daha güçlü olduğu gösterilmesi gerekmektedir. İlk S-box çıktısını hassas veri olarak kullanarak üzerinde korelasyon hesaplaması tekrarlanmıştır. Hassas verilerde azalma olmasına rağmen Şekil 48'de gösterildiği gibi veriler ve güç izi değerleri arasında daha güçlü korelasyon elde edilmiştir.



Şekil 48. Güç izindeki her j noktası için ilk tur hassas verileri ile güç izleri arasındaki daha güçlü oluşan korelasyon.

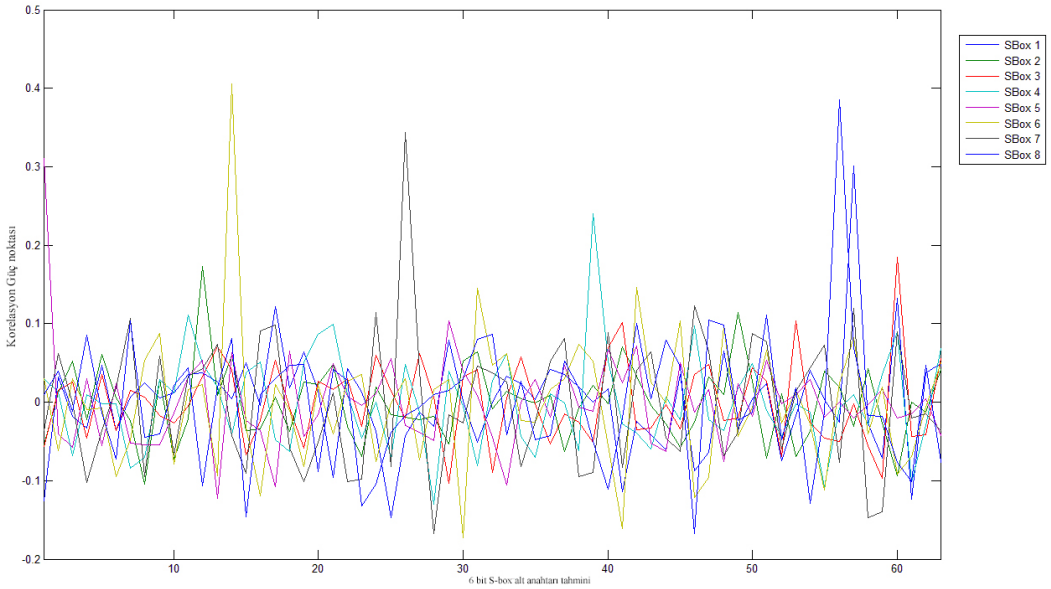
Yapılan bu işlemlere ek olarak DES şifrelemesine saldırıyı tam anlamıyla gerçekleştirilmesi için kullanılan anahtar bitleri tahmin edilmeli ve S-box için en

yüksek korelasyon seçilmelidir. Her S-box için, Feistel işlevinde kullanılan düz metinden 6 bit seçilmiştir. Örneğin, ilk S-box durumunda bu bitler düz metnin (7, 57, 49, 41, 33 25)'dir. S-box'ın çıktısı bilinmeyen 6 bitlik anahtara bağlı olduğundan, her anahtar tahmini için bir tane (0'dan 63'e kadar) olmak üzere sadece 26 potansiyel S-box çıktı dizisi üretilmiştir. 4 bitlik S-box çıkışı için, S-box çıktısı ile düz metindeki karşılık gelen bitler arasındaki Hamming mesafesi hesaplanmıştır. Örnek olarak anahtar tahmini 12 yi seçersek $HW[R2_{12}(1:4) \oplus R1(1:4)]$ şeklinde hesaplanır. Burada $R2_{12}$ anahar tahmini 12'nin potansiyel değeri olarak kabul edilmiştir. Her potansiyel anahtar 64 hassas veri değerinden oluşmaktadır. Korelasyon katsayısı denklemi kullanılarak, 64 hassas veri değerinin her biri ile güç bilgisi arasındaki korelasyonu hesaplanmıştır. Bu işlem birçok güç izi kullanılarak korelasyon katsayısı oluşturulması için tekrarlanmıştır. Yeteri kadar güç izinden sonra korrelasyonda doğru anahtara karşılık gelen bir tepe noktası oluşmuştur. 500 güç izinden elde edilen tek bir S-box kutusuna ait sonuçlar Şekil 49'da gösterilmektedir. Şekilden görüldüğü gibi tahmin 14'te korelasyonda artış olduğu doğru anahtar tahmini olduğunu göstemektedir. Böylece S-box içinde doğru alt anahtarın 13 olduğu belirtilmektedir (Şekil 49).



Şekil 49. Tek bir S kutusundaki her 6 bitlik anahtar tahmini için güç bilgisi ile hassas veriler arasındaki korelasyon.

Bu işlemi 8 bitlik S-box için 64 hassas veri değerini içeren bir korelasyon tablosu oluşturmak üzere her S-box için tekrarlanır. Her S-box için doğru tahmin doğru anahtarı göstermektedir. Tüm 8 bitlik S-box için hassas verilerin tüm örnekleri Şekil 50'de gösterilmektedir. Şekildeki korelasyon grafiğindeki tepe nokataları olarak ilk tur anahtarı (56 11 59 38 0 13 25 55) için doğru 6 bitlik anahtar değerlerini göstermektedir.



Şekil 50. 8 S-box'ın tümü için hassas veriler ve güç bilgileri arasındaki korelasyon.

Program belirli sayıda güç izi için bir anahtara karar vermektedir.verilen karar ile program MATLAB konsuluna alt anahtarı, belirlemek için geçen süreyi ve doğru alt anahtar ile anahtar belirlemek için gerekli olan güç izi sayısını vermektedir. Program çıktısının bir örneği Şekil 51'de gösterilmektedir. Program DPA saldırsında verirabında 800 güç izinden yaklaşık 455 iz kullanarak alt anahtar elde edilebilmiştir. Yapılan denemelerde anahtarı bulmak için maksimum 826, minimum 269 güç izine ihtiyaç duyulmuştur. Ortlama olarak 485 güç izinden yararlanarak anahtar bulmuşlardır. DPA saldırsında daha gelişmiş girişler ve daha gelişmiş istatiksel yöntemler sayesinde güç izi sayısı 141 güç izine düşürülebilmişlerdir. Yapılan saldırılarda alt anahtarın bulunması ise 3 ile 4 dakika sürmüştür (Hnath and Pettengill, 2010).

```
>> cd C:\secmatv1_2006_04_0809\
Cracking DES, standby...
Iteration: 1      # Of Traces Required: 383      Trace Number: 383
Iteration: 2      # Of Traces Required: 451      Trace Number: 833
Iteration: 3      # Of Traces Required: 610      Trace Number: 1442
Iteration: 4      # Of Traces Required: 444      Trace Number: 1885
Iteration: 5      # Of Traces Required: 642      Trace Number: 2526
Iteration: 6      # Of Traces Required: 374      Trace Number: 2899
Iteration: 7      # Of Traces Required: 315      Trace Number: 3213
Iteration: 8      # Of Traces Required: 302      Trace Number: 3514
Iteration: 9      # Of Traces Required: 320      Trace Number: 3833
Iteration: 10      # Of Traces Required: 430      Trace Number: 4262
Iteration: 11      # Of Traces Required: 675      Trace Number: 4936
Iteration: 12      # Of Traces Required: 539      Trace Number: 5474
Iteration: 13      # Of Traces Required: 474      Trace Number: 5947
Iteration: 14      # Of Traces Required: 384      Trace Number: 6330
Iteration: 15      # Of Traces Required: 398      Trace Number: 6727
Iteration: 16      # Of Traces Required: 610      Trace Number: 7336
Iteration: 17      # Of Traces Required: 407      Trace Number: 7742
Iteration: 18      # Of Traces Required: 417      Trace Number: 8158
Iteration: 19      # Of Traces Required: 269      Trace Number: 8426
Iteration: 20      # Of Traces Required: 331      Trace Number: 8756
??? Operation terminated by user during ==> DES_attack at 157
```

Şekil 51. DES DPA saldırı programından örnek çıktı.

6. SONUÇ

Tez kapsamında akıllı kartlarda kullanılan ve akıllı kartların güvenliğini sağlayan DES ve AES şifreleme sistemleri üzerinde yapılmış uygulamalar sunulmuştur. Bu uygulamalarda farklı yöntemler ile akıllı kartlarda kullanılan şifreleme algoritmalarının alınan önlemlere rağmen anahtarlarına ulaşılabildiği anlatılmıştır. Sonuçlar dikkate alındığında, AES algoritması DES algoritmasına göre çok daha güvenilir bir şifreleme sistemidir. Hnath and Pettengill yaptığı saldırı denemelerinde de elde ettiği gibi DES şifreleme sistemini çözmek için 455 güç izine ve ortalama 3-4 dakikalık (Hnath and Pettengill, 2010) bir süreye ihtiyaç duyarken AES şifreleme sistemini çözmek için 10000 güç izine ve ortalama 9 dakikalık bir süreye ihtiyaç duymuştur.

AES şifrelemesinin döngü anahtarı ve sütun karıştıma aşamalarının şifrelemeyi daha karmaşık hale getirmesi şifrelemenin deşifresini zorlaştırmaktadır. Ayrıca AES şifrelemesine uygulanan maskeleme sayesinde deşifre edilmesi daha da zorlaşmıştır. Böylece AES şifrelemesi basit güç saldırılarına, şablon saldırlarına ve birinci dereceden DPA saldırılarına karşı korunabilmektedir. Fakat AES şifrelemesi Oswald et al. tarafından gerçekleştirilen ikinci dereceden DPA saldırısına karşı kendini koruyamamıştır Çünkü iki dereceden DPA saldırısında karmaşıklığın değerlendirilmesi kolaylaşmaktadır. Bu sayede ikinci dereceden DPA saldırısının maskeli yazılımlar için bir tehdit olduğu gösterilmiştir (Oswald et al., 2006).

AES ve DES şifrelemeleri için en önemli etkenlerden biri de gürültüdür. Gürültü sayesinde alınan güç izlerinde anahataın elde edilmesi daha da zorlaşmaktadır. Çünkü gürültü, elde edilen güç izlerine ait sinyallere etki

etmektedir. Bu da anahtarın tespit edilmesini sağlayan tepe noktasının tespitini zorlaştırmaktadır. Fakat sinyal-gürültü oranın iyileştirilmesi için güvenilir bir kurulum ve korelasyon katsayısı ile gürültüler gerçek sinyallerden ayırt edilmektedir. Örs et.al. yaptıkları saldırıda korelasyon analizi sayesinde yonga test cihazının önemli bir miktarda gürültüye sebep olduğunu tespit etmişlerdir (Örs et al., 2005).

Akıllı kartlarda kullanılan şifreleme sistemlerinin zayıf olmasının temel nedeni akıllı kartların 8 bitlik bir şifrelemeye sahip olmasıdır. Çünkü akıllı kartlarda bulunan mikroişlemci yongalarının hafızaları düşüktür. Eğer işlemci hafızaları daha yüksek olsaydı kullanılan bit sayısının daha yüksek olabileceği düşünülmektedir. Bu sayede akıllı kartların kriptosistemlerinin deşifre edilmesi de giderek zorlaşabilir. Fakat mevcut teknolojide akıllı kartlarda kullanılan AES ve DES algortimalarının yan kanal saldırılarıyla kriptosistemlerinin kırılabileceği çok açıktır. Ancak yan kanal saldırısını gerçekleştirmek için gereken en önemli etken saldırı yapılacak olan akıllı kartın fiziksel donanımına erişimin mümkün olmasıdır.

Akıllı kartların güvenliği günümüzde yaşanan pandemi nedeniyle giderek önem kazanmıştır, çünkü insanların temastan kaçınmak için temassız kart kullanım eğilimi bu dönemde artmıştır. Temassız kartların deşifre edilmesinin günümüz teknolojisinde mümkün olmadığı görülmektedir. Çünkü deşifre edilmesi için temassız karta fiziksel erişimin sağlanması gerekmektedir. Fakat ilerleyen teknolojide eğer akıllı kartlarda kullanılan işlemci hafızası artırılmaz ise gelişen elektronik devre ve bilgisayar teknolojisi sayesinde akıllı kartlarda kullanılan kriptosistemlerin deşifre edilmesinin giderek kısa sürede çözülebileceği tahmin edilmektedir.

KAYNAKLAR DİZİNİ

Aslan, F.Y., Sakallı, M.T. ve Aslan, B., 2012, Önemli Blok Şifrelerde Kullanılan Doğrusal Dönüşümlerin İncelenmesi. *Akademik Bilişim'12,* 46-56 s.

Aydoğan, S.S., 2016. Mukayeseli Veri Şifreleme Algoritmaları, Seminer Raporu, Yıldız Teknik Üniversitesi, Bilgisayar Mühendisliği Bölümü, 5s (yayımlanmamış).

Herbst, C., Oswald, E. and Mangard, S., 2006, An AES Smart Card Implementation Resistant to Power Analysis Attacks, 239-252, Applied Cryptography and Network Security, J. Zhou, M. Yung and F. Bao (Eds.), 4th International Conference, ACNS 2006, 485p.

Çimen, C., **Akyıldız, E. ve Akleylek, S.**, 2009, Şifrelerin Matematiği: Kriptografi, ODTÜ Yayıncılık, Ankara, 137s.

Di Natale, G., **Doulcier, M., Flottes, M.L. and Rouzeyre, B.**, 2009, A Reliable architecture for parallel implementations of the advanced encryption standard. *J. Electron. Test. Theory Appl,.* 25: 269–278 pp.

Dirk, H., 2001, Standards in the smart card world, *Comput. Networks,* 36: 473–487 pp.

Büyükkaya E., 2017, Raspberry Pi üzerinde AES algoritmasına Yan Kanal Analizi ve Ölçüm İyileştirme, Yüksek Lisans Tezi, İstanbul Şehir Üniversitesi Fen Bilimleri Enstitüsü Bilgi Güvenliği Mühendisliği Ana Bilim Dalı, 76s (yayımlanmamış).

Gamaarachchi, H. and Ganegoda, H., 2018, Power Analysis Based Side Channel Attack, CO411/2::Individual Project I & II – Report, University of Peradeniya, 101p (unpublished).

KAYNAKLAR DİZİNİ (devam)

Grabbe, J., 1992, The DES algorithm illustrated, *Laissez Faire City Times,* 2(28): 1–15 pp.

Hatun, E., 2018, Raspberry Pi üzerinde gerçeklenmiş RSA algoritmasına yan kanal analizi, Yüksek Lisans Tezi, İstanbul Şehir Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği Ana Bilim Dalı, 83s (yayımlanmamış).

Hnath, W. **and Pettengill, J.**, 2010, Differential Power Analysis Side-Channel Attacks in Cryptography, Worcester Polytechnic Institute, 42p (unpublished).

Kang, S. M. ve Leblebici, Y., 1983, CMOS Digital Integrated Circuits, 15th *Conference on Solid State Devices and Materials*. 61-64 pp.

Keyman, E. ve Yıldırım, M., 2004, Kriptolojiye Giriş, ODTÜ Uygulamalı Matematik Enstitüsü, Ankara, 148s.

Kocher, **P.C.**, 1996, Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems., 104-113, Advances in Cryptology- CRYPTO ’96, Lecture Notes in Computer Science, N. Koblitz (Ed.), vol 1109. Springer, 417p.

Kocher, P., Jaffe, J. and Jun, B., 1999, Differential Power Analysis, 388–397, Advances in Cryptology - CRYPTO '99, M. Wiener (Ed.), Springer, 638p.

Le, T.H., Clédière, J., Servière, C. and Lacoume, J.L., 2007, Noise reduction in side channel attack using fourth-order cumulant, *IEEE Trans. Inf. Forensics Secur.* 2(4), 710–720pp.

KAYNAKLAR DİZİNİ (devam)

Ordu, L. ve Örs Yalçın S.B. "Yan-Kanal Analizi Saldırılarına Genel Bakış", https://kamusm.bilgem.tubitak.gov.tr/dosyalar/makaleler/Yan-Kanal%20Analizi%20Saldirilarina%20Genel%20Bakis.pdf (Erişim tarihi 15 Temmuz 2021).

Lo, O., Buchanan, W.J. and Carson, D., 2017, Power analysis attacks on the AES-128 S-box using differential power analysis (DPA) and correlation power analysis (CPA), *J. Cyber Secur. Technol.* 1, 88–107pp.

Mangard, S., Oswald, E. and Popp, T., 2007, Power Analysis Attacks: Revaling the Secrets of Smart Cards, Springer, 281p.

Mayes, K. and Markantonakis, K., 2017, Smart Cards, Tokens, Security and Applications, Springer, 379p.

Ordu, L., 2006, AES algoritmasının FPGA üzerinde gerçeklenmesi ve yan kanal analizi saldırılarına karşı güçlendirilmesi, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği Ana Bilim Dalı, 107s (yayımlanmamış).

Örs, S.B., Gürkaynak, F.K., Oswald, E. and Preneel, B., 2004, Power-analysis attack on an ASIC AES implementation, *International Conference on Information Technology: Coding and Computing*, 2:546-552pp.

Oswald, E., Mangard, S., Herbst, C. and Tillich, S., 2006, Practical Second-Order DPA Attacks For Masked Smart Card Implementations Of Block Ciphers, 192-207, Topics in Cryptology – CT-RSA, Lecture Notes in Computer Science, D. Pointcheval (Ed.), vol 3860, Springer, 364p.

KAYNAKLAR DİZİNİ (devam)

Rothke, B., 2007, A look at the Advanced Encryption Standard (AES), 1151-1158, Inf. Secur. Manag. Handbook, 6th Edition, 1151–1158, H.F. Tipton and M. Krause (Eds.), CRC Press, 3280p .

Şahin, F., 2015, Modern Blok Şifreleme Algoritmaları, *İstanbul Aydın Üniversitesi Derg.* 26: 23–40 s.

Shannon, C.E., 1949, Communication Theory of Secrecy Systems, *Bell Syst. Tech. J.* 28: 656–715.

Shelfer, K.M. and Procaccino, J.D., 2002, Smart card evolution, *Commun. ACM* 45, 83–88. https://doi.org/10.1145/514236.514239

Standaert, F.X., Rouvroy, G. and Quisquater, J., 2006, FPGA implementations of the des and triple-des masked against power analysis attacks, *Proc. - 2006 Int. Conf. F. Program. Log. Appl. FPL* 1:791–794 pp.

Tiu, C.C., 2005, A New Frequency-Based Side Channel Attack for Embedded Systems, MSc Thesis, University of Waterloo, 90p (unpublished).

Yıldırım K., ve Demiray, H.E., 2008, Simetrik Ve Asimetrik Şifreleme Yöntemlerine Metotlar: Çirpilmiş Ve Birleşik Akm-Vkm, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 23(3): 539-548 s.

TEŞEKKÜR

Tez çalışmam boyunca, bilgi ve tecrübelerini benden esirgemeyen ve her türlü desteği göstererek yanımda olan danışman hocam sayın Dr. Öğr. Üyesi Arif GÜRSOY'a, bilgi birikiminden ve deneyimlerinden çok şey öğrendiğim sayın Prof. Dr. Urfat NURİYEV'e ve sayın Dr. Öğr. Üyesi Erdem ALKIM'a, desteğini her zaman hissettiğim aileme ve sevgili nişanlım Dr. Güliz AK'a, ayrıca 2210-C Öncelikli Alanlara Yönelik Yüksek Lisans Burs Programı ile tez çalışmam boyunca maddi destek veren TÜBİTAK-BİDEB'e sonsuz teşekkürlerimi sunarım.

ÖZGEÇMİŞ

Lise eğitimini Bodrum Lisesi'nde tamamladı. 2008 yılında Ege Üniversitesi Fizik Bölümünü kazandı. 2018 yılında Ege Üniversitesi Fen Bilimleri Enstitüsü Matematik Anabilim Dalı'nda yüksek lisans öğrenimine başladı.

BİLDİRİLER

Ulusal bilimsel toplantılarda sunulan ve bildiri kitabında basılan bildiriler

- Karatay M., Demiroz D., Alkım E., Gürsoy A., "Kriptografik İmzalamada Bazı Özet Fonksiyonların Karşılaştırılması", Sözlü Bildiri, Uluslararası Marmara Fen ve Sosyal Bilimler Kongresi (Bahar 2019), Kocaeli, 26 – 28 Nisan 2019.

DÜZENLENEN BİLİMSEL TOPLANTILAR (SEMİNERLER)

- Akıllı kimlik kartlarının finansal işlemlerde kullanımı: olası güvenlik tehditleri ve alınacak önlemler, Ege Üniversitesi Fen Fakültesi Matematik Bölümü, 25 Aralık 2018.

KAZANILAN BURSLAR

- TÜBİTAK 2210-C Öncelikli Alanlara Yönelik Yurt İçi Yüksek Lisans Bursu 2019/1

KATILINAN BİLİMSEL ETKİNLİKLER

- İzmir Genç Fizikçiler Kongresi 2012, Ege Üniversitesi, İzmir, 13-15 Haziran 2012.
- Temel Bilimler Sempozyumu 2014 (Organizasyon Komitesi), İzmir, 15 Mayıs 2014.
- İzmir Genç Fizikçiler Kongresi 2013, Dokuz Eylül Üniversitesi, İzmir, 8-10 Temmuz 2013.
- İzmir Genç Fizikçiler Kongresi 2016 Ege Üniversitesi, İzmir, 12-14 Ekim 2016.
- CyberEge'18-Siber Güvenlik Etkinliği ve Eğitimi, Ege Üniversitesi, İzmir, 12-13 Mayıs 2018.
- TBD Uluslararası Biyokimya Kongresi / 29. Ulusal Biyokimya Kongresi, Muğla 26-30 Ekim 2018.
- Microsoft Eğitim Çözümleri Paneli, Ege Üniversitesi, İzmir, 25 Şubat 2019
- Sağlıkta Yapay Zeka Uluslararası Sempozyumu (International Symposium on Artificial Intelligence in Healthcare), Ege Üniversitesi Tıp Fakültesi, İzmir, 7-8 Şubat 2020.
- 22. Ege Onkoloji Günleri – Onkoloji Kursu / Onkolojide Yeni Tedavi arayışları Sempozyumu, Ege Üniversitesi Tıp Fakültesi, İzmir, 26-28 Şubat 2020.
- Uzaktan eğitim, iş ve kariyer - 5n1k - çevrimiçi konferansı, Muğla Üniversitesi, Muğla, 08-10 Mayıs 2020.
- Online Yapay Zeka Uygulamalari Workshop, Bahçeşehir Üniversitesi, İstanbul, 13 Haziran 2020.

- 14. İstanbul Bilişim Kongresi, Türkiye Bilişim Derneği, İstanbul, 9-10 Aralık 2020.

SERTİFİKALAR

- Adobe Photoshop CC 2020, 29 Nisan 2020
- Uygulamalı Destekli Kriptoloji Eğitimi, 04 Mayıs 2020
- C# ile Kriptoloji, 05 Mayıs 2020
- Temel Programlamaya Giriş, 10 Mayıs 2020
- Dijital Pazarlamanın Temelleri, 16 Mayıs 2020
- Html5 ve CSS3 Eğitimi, 17 Mayıs 2020
- Adobe After Effect Eğitimi, 17 Mayıs 2020
- SPSS ile İstatistik Eğitimi, 19 Mayıs 2020
- Programlamanın Temelleri, 22 Mayıs 2020
- Arduino ile Mikrodenetleyicilere Giriş, 24 Mayıs 2020
- Wordpresss Eğitimi, 24 Mayıs 2020
- Sosyal Medya Uzmanlığı Eğitimi, 24 Mayıs 2020
- Temel Elektronik, 25 Mayıs 2020
- Digital Signal Processing with MATLAB, 30 Mayıs 2020
- İleri Excel Eğitimi, 07 Haziran 2020
- Pyhton Eğitimi, 07 Haziran 2020
- Adobe Premiere Temel Eğitimi, 07 Haziran 2020

- A'dan Z'ye Uygulamalı Microsoft Office Programları Eğitimi,06 Ağustos 2020
- Yapay Zeka: Python ile Programlama,05 Ocak 2021
- Bilgi Güvenliği, 05 Ocak 2021
- Sızma Testi, 09 Ocak 2021