T.C.
İSTANBUL ÜNİVERSİTESİ-CERRAHPAŞA
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

NESNELERİN İNTERNETİNDEKİ DONANIMLAR İÇİN YAN KANAL SALDIRILARINA KARŞI SİBER GÜVENLİK ANALİZİ

Mustafa ŞİRİN

DANIŞMAN
Prof. Dr. Fırat KAÇAR

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Elektrik-Elektronik Mühendisliği Programı

Aralık, 2022

TEZ KABUL VE ONAYI

Mustafa ŞİRİN tarafından, **Prof. Dr. Fırat KAÇAR** danışmanlığında hazırlanan **"NESNELERİN İNTERNETİNDEKİ DONANIMLAR İÇİN YAN KANAL SALDIRILARINA KARŞI SİBER GÜVENLİK ANALİZİ"** başlıklı bu çalışma, jürimiz tarafından **14/12/2022** tarihinde yapılan sınav sonucunda **oy birliği** ile başarılı bulunarak **Yüksek Lisans Tezi** olarak kabul edilmiştir.

Tez Jürisi

		İmza	**Sonuç**
DANIŞMAN	Prof. Dr. Fırat KAÇAR İstanbul Üniversitesi – Cerrahpaşa Elektrik-Elektronik Mühendisliği Anabilim Dalı		☒ Kabul ☐ Ret
ÜYE	Doç. Dr. Muhammed Ali AYDIN İstanbul Üniversitesi – Cerrahpaşa Bilgisayar Mühendisliği Anabilim Dalı		☒ Kabul ☐ Ret
ÜYE	Doç. Dr. Şerif BAHTİYAR İstanbul Teknik Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı		☒ Kabul ☐ Ret

Aileme ithaf ediyorum…

BÜTÇE DESTEKLERİ

NESNELERİN İNTERNETİNDEKİ DONANIMLAR İÇİN YAN KANAL SALDIRILARINA KARŞI SİBER GÜVENLİK ANALİZİ

Bu tez çalışması için herhangi bir kurumdan bütçe desteği alınmamıştır.

TEŞEKKÜR

Hem lisans hayatımda hem de yüksek lisans hayatımda desteğini esirgemeyen, beni donanım güvenliği konusuna yönlendirerek bu konuda bir projede çalışmama vesile olan ve hazırladığım tez çalışması süresince beni yönlendirip destek olan değerli hocam Prof. Dr. Fırat KAÇAR'a, Nesnelerin İnterneti siber güvenliği konusunda yer aldığım projede başta Doç. Dr. Muhammed Ali AYDIN ve Araş. Gör. Ebu Yusuf GÜVEN olmak üzere tüm proje ekibime teşekkürlerimi sunarım. Son olarak beni tüm eğitim ve kariyer hayatım boyunca destekleyen ve verdiğim kararların arkasında duran canım aileme sonsuz teşekkürlerimi sunarım.

Aralık 2022 **Mustafa ŞİRİN**

İÇİNDEKİLER

Sayfa No

TEZ KABUL VE ONAYI ..... **ii**
BEYAN ..... **iii**
BÜTÇE DESTEKLERİ ..... **v**
TEŞEKKÜR ..... **vi**
İÇİNDEKİLER ..... **vii**
ŞEKİL LİSTESİ ..... **ix**
TABLO LİSTESİ ..... **xiii**
SİMGE VE KISALTMA LİSTESİ ..... **xiv**
ÖZET ..... **xvi**
ABSTRACT ..... **xviii**
1. GİRİŞ **1**
2. KAVRAMSAL ÇERÇEVE **3**
2.1. NESNELERİN İNTERNETİ VE GÜVENLİĞİ 3
2.2. DONANIM GÜVENLİĞİ 12
2.2.1. Donanım Truva Atı 17
2.2.2. Firmware Saldırıları 17
2.2.3. Yan Kanal Saldırıları 19
2.2.4. Donanım Güvenliği Saldırı Örnekleri 20
2.3. KRİPTOGRAFİ 21
3. YÖNTEM **26**
3.1. YAN KANAL SALDIRILARI 26
3.1.1. Güç Analizi Saldırıları 28
3.1.2. Zamanlama Analizi 32
3.2. TEST ORTAMI 33
3.2.1. Chipwhisperer 35
3.2.2. ATmega328P Mikrodenetleyicisi 36
3.2.3. ATXMEGA128D4-AU Mikrodenetleyicisi 37
3.2.4. Test Senaryoları 38

4. BULGULAR **40**
4.1. PAROLA KORUMALI SİSTEME ZAMANLAMA ANALİZİ SALDIRISI 40
4.1.1. ATmega328P Mikrodenetleyici Üzerinde Parolanın Karakter Olarak Karşılaştırılması 40
4.1.2. ATmega328P Mikrodenetleyici Üzerinde Parolanın Kelime Olarak Karşılaştırılması 43
4.1.3. ATmega328P Mikrodenetleyici Üzerinde Parolanın Hash Değerinin Karşılaştırılması 47
4.1.4. ATmega328P Mikrodenetleyici Üzerinde Parola Karşılaştırma Yapısına Rastgele Zaman Gecikmesi Koyularak Parolanın Kontrol Edilmesi 50
4.1.5. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parolanın Karakter Olarak Karşılaştırılması 52
4.1.6. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parolanın Kelime Olarak Karşılaştırılması 55
4.1.7. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parolanın Hash Değerinin Karşılaştırılması 57
4.1.8. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parola Karşılaştırma Yapısına Rastgele Zaman Gecikmesi Koyularak Parolanın Kontrol Edilmesi 60
4.2. RSA ŞİFRELEME GERÇEKLEŞTİREN DONANIMDA GÜÇ ANALİZİ SALDIRISI 62
4.2.1. Basit Güç Analizi ile XMEGA128D4-AU Mikrodenetleyicisi Üzerinden Gizli Anahtarın Elde Edilmesi 62
4.2.2. XMEGA128D4-AU Mikrodenetleyici Üzerinde RSA Şifreleme Adımlarına Anlamsız Rastgele Döngülerin Eklenmesi Durumunda Basit Güç Analizi 63
5. TARTIŞMA **68**
6. SONUÇ VE ÖNERİLER **73**
KAYNAKLAR **77**
İNTİHAL RAPORU İLK SAYFASI **82**
ETİK KURUL İZİN YAZISI **83**
KURUM İZNİ YAZILARI **84**
ÖZGEÇMİŞ **85**

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: IoT yapısı ağların ağı şeklinde görülebilir [7]. 4

Şekil 2.2: IoT'nin basitleştirilmiş yapısı [8]. 5

Şekil 2.3: Basit şekilde bir nesnenin yapısı. 6

Şekil 2.4: Endüstri pazarında giyilebilir IoT çözümleri tarafından popüler olarak hedeflenen farklı vücut bölümleri [5]. 8

Şekil 2.5: Çevremizde siber saldırıya uğrayabilecek cihazlar [21]. 9

Şekil 2.6: Nesnelerin İnternetinde Güvenlik [26]. 11

Şekil 2.7: IoT cihazlarında güvenlik açıkları ve etkileri piramidi [28]. 12

Şekil 2.8: Donanım tasarım ve test süreci akışındaki adımlar ve sızıntı olabilecek noktalar [29]. 14

Şekil 2.9: Bir çipin yaşam döngüsündeki her aşama için saldırı vektörleri ve karşı önlemleri [29]. 14

Şekil 2.10: Simetrik Şifreleme. Alice M mesajını K anahtarını kullanarak E algoritmasıyla şifreleyip C şifrelenmiş metnini Bob'a gönderiyor. Bob aynı K anahtarını kullanarak C şifreli metninin şifresini çözerek M mesajını elde ediyor [42]. 22

Şekil 2.11: Açık Anahtarlı Şifreleme. Asimetrik Şifreleme. Alice M mesajını Bob'a ait 'public' anahtarını (pk) kullanarak E şifreleme algoritmasıyla şifreleyip C şifreli metnini Bob'a gönderiyor. Bob, C şifreli metninin kendine ait olan 'private' anahtar (sk) ile şifresini çözüp M mesajını elde ediyor [42]. 23

Şekil 3.1: Kriptografik donanımın çalışması esnasında ortaya çıkan yan kanal sızıntıları [29]. 26

Şekil 3.2: Bir CMOS eviricinin durum değişimleri [29]. 29

Şekil 3.3: Bir CMOS eviricinin anahtarlama esnasında dinamik güç tüketimi [47]. 29

Şekil 3.4: Kriptografik işlem gerçekleştiren bir donanıma yapılan güç analizi saldırısının kurulumu [29]. 30

Şekil 3.5: RSA şifre çözme işleminde modüler üs alma güç izlerinin bir bölümü [29]. 31

Şekil 3.6: Zamanlama analizi saldırılarının kavramsal bir görünümü [50]. 32

Şekil 3.7: Lojik analizör ile hedef donanım üzerinden işlem süresi ölçümü için oluşturulan test ortamı. 34

Şekil 3.8: Mikrodenetleyicinin güç ölçümü için oluşturulan test ortamı. 35

Şekil 3.9: Tez çalışmasında kullanılan Chipwhisperer-Lite aracı. 36

Şekil 3.10: ATmega328P mikrodenetleyicisi üzerinden güç ölçümü kurulumu. 37

Şekil 3.11: ATmega328P mikrodenetleyicisinin güç analizi ve zamanlama analizi için kurulan test ortamı. 37

Şekil 3.12: ATXMEGA128D4-AU mikrodenetleyicisi için hazırlanan test ortamı. 38

Şekil 4.1: Arduino içerisinde yapılan parolanın karakter olarak karşılaştırılması işlemi. 40

Şekil 4.2: Seri port üzerinden girilen hatalı parola sonrası karşılaşılan hata uyarısı. 41

Şekil 4.3: ‘a’ ve ‘h’ harflerinin zaman gecikmesi 41

Şekil 4.4: ‘ha’ ve ‘h0’ girdilerinin zaman gecikmesi. 42

Şekil 4.5: ‘a’ ve ‘h’ harflerinin işlem esnasında harcadıkları gücü gösteren mikrodenetleyiciye ait güç sinyallerinin osiloskop görüntüleri. 43

Şekil 4.6: Arduino içerisinde yapılan parolanın kelime olarak karşılaştırılması işlemi. 43

Şekil 4.7: 'a' harfi ve 'h' harfine ait zaman gecikmesi. 44

Şekil 4.8: 'aa', 'aaa', 'aaaa' ve 'aaaaa' girdileri sonrası ölçülen işlem süreleri. 44

Şekil 4.9: 'baaa' ve 'haaa' girdileri sonrası ölçülen işlem süreleri. 45

Şekil 4.10: 'a', 'h', 'aaaa' ve 'haaa' girdilerinin işlem esnasında harcadıkları gücü gösteren mikrodenetleyiciye ait güç sinyallerinin osiloskop görüntüleri. 47

Şekil 4.11: Açık metin halinde girilen parolanın hash değerinin alınıp sistemin hash değeriyle karşılaştırılması. 47

Şekil 4.12: 'b' girdisinin ilk süre ölçümü (solda) ve ikinci süre ölçümü (sağda). 48

Şekil 4.13: 'aaaa' ve 'aaaaa' girdilerinin işlem süreleri. 48

Şekil 4.14: 'a' girdisi ile sisteme yapılan iki denemeye ait güç sinyali 49

Şekil 4.15: Sisteme 3 defa 'h' girdisi yapıldığında elde edilen 3 farklı işlem süresi. 50

Şekil 4.16: 'a', 'c' ve 'h' girdisi için elde edilen mikrodenetleyicinin güç profilleri. 51

Şekil 4.17: 'a', 'c' ve 'h' girdileri için elde edilen mikrodenetleyicinin güç profilleri 52

Şekil 4.18: XMEGA üzerinde yapılan parolanın karakter karakter karşılaştırılarak doğrulama yapılması. 52

Şekil 4.19: 'a' ve 'h' girdileri sonrası mikrodenetleyici üzerinden lojik analizör ile ölçülen işlem süreleri. 53

Şekil 4.20: 'ha' ve 'h0' girdileri sonrası mikrodenetleyici üzerinde lojik analizör ile elde edilen işlem süreleri. 53

Şekil 4.21: 'a' ve 'h' girdileri sonrası osiloskop ile elde edilen mikrodenetleyiciye ait güç profilleri 54

Şekil 4.22: Parola doğrulaması için yapılan girdi ve sistem parolasının kelime olarak karşılaştırılması. 55

Şekil 4.23: 'a' ve 'h' girdilerinin lojik analizör ile ölçülen işlem süreleri. 55

Şekil 4.24: 'a' ve 'aaaa' girdileri için lojik analizör ile ölçülen işlem süreleri. 56

Şekil 4.25: Mikrodenetleyiciye yapılan 'a' ve 'h' girdileri sonrası osiloskop ile elde edilen güç profilleri. 57

Şekil 4.26: Parolaların hash değerlerinin karşılaştırılarak parola doğrulaması yapılması. 57

Şekil 4.27: 'a', 'b' ve 'h' girdileri için lojik analizör ile elde edilen işlem süreleri. 58

Şekil 4.28: Aynı karakter uzunluğuna sahip girdiler sonrasında elde edilen farklı işlem süreleri 59

Şekil 4.29: 'h' girdisinin iki farklı zamanda denenmesiyle mikrodenetleyici üzerinden elde edilen güç profili. 60

Şekil 4.30: 'he' girdisinin farklı zamanlarda girilmesi sonrası elde edilen işlem süreleri. 60

Şekil 4.31: 'a', 'd' ve 'h' girdileri sonrası ekrana uyarı mesajı çıkma anındaki mikrodenetleyiciye ait güç profili. 61

Şekil 4.32: 'a', 'd' ve 'h' girdileri yapıldıktan hemen sonra elde edilen mikrodenetleyiciye ait güç profili. 61

Şekil 4.33: XMEGA üzerinde RSA şifre çözme adımına ait güç izi. 62

Şekil 4.34: RSA şifre çözme adımına ait güç izinin alçak geçiren filtre uygulanmış hali. 63

Şekil 4.35: Şifreleme algoritmasına ait kodların içerisine eklenebilecek anlamsız kod parçacığı örneği. 64

Şekil 4.36: Anlamsız kod parçacığının eklendiği sistemde şifre çözme adımı başladığı andan itibaren yakalanan güç izi. 64

Şekil 4.37: Şekil 4.36'da yer alan güç izinin alçak geçiren filtre uygulanmış hali. 65

Şekil 4.38: Şifreleme algoritmasına anlamsız kod parçacığı eklenmiş sistemin güç izi. 65

Şekil 4.39: Şekil 4.38'de yer alan güç izinin alçak geçiren filtre uygulanmış hali. 66

Şekil 4.40: Anlamsız kod parçacığındaki döngü tur sayısının azaltılmasıyla elde edilen güç izi. 67

Şekil 4.41: Şekil 4.40'ta yer alan güç izinin alçak geçiren filtre uygulanmış hali. 67

TABLO LİSTESİ

Sayfa No

Tablo 2.1: Donanımı ilgilendirebilecek saldırı yüzeyleri ve güvenlik açıkları [32].....16

Tablo 4.1: Seri port üzerinden yapılan girdiler ve lojik analizör ile ölçülen zaman gecikmeleri 42

Tablo 4.2: Seri port üzerinden yapılan girdiler ve işlem süreleri. 46

Tablo 4.3: Seri port üzerinden yapılan girdiler ve girdilere ait işlem süreleri.....49

Tablo 4.4: Lojik analizör ile farklı girdilerin 3 ayrı ölçümünden elde edilen işlem süreleri..51

Tablo 4.5: Sisteme yapılan parola tahmin girdileri ve girdilerin işlem süreleri. 54

Tablo 4.6: Parolayı elde etmek için sisteme yapılan girdiler ve işlem süreleri. 56

Tablo 4.7: Mikrodenetleyici üzerinde yapılan girdiler ve ölçülen işlem süreleri.....59

SİMGE VE KISALTMA LİSTESİ

Simgeler **Açıklama**

B : Byte
KB : KiloByte
ms : milisaniye
nF : nanofarad
V : Volt
µs : mikrosaniye

Kısaltmalar **Açıklama**

ADC : Analog-to-Digital Converter
AES : Advanced Encryption Standard
BGA : Ball Grid Array
CMOS : Complementary Metal Oxide Semiconductor
CPA : Correlation Power Analysis
DDoS : Distributed Denial-of-Service
DES : Data Encryption Standard
DNS : Domain Name System
DPA : Differential Power Analysis
EEPROM : Electronically Erasable Programmable Read-Only Memory
FPGA : Field-Programmable Gate Array
GND : Ground
GPU : Graphics Processing Unit
I/O : Input / Output
I2C : Inter-Integrated Circuit
ID : Identification
IoT : Internet of Things
IP : Intellectual Property
JTAG : Joint Test Action Group
MCU : Microcontroller Unit

MD5 : Message-Digest algorithm 5
MIT : Massachusetts Institute of Technology
NIST : National Institute of Standards and Technology
OWASP : Open Web Application Security Project
P&G : Procter & Gamble
PC : Personal Computer
PCB : Printed Circuit Board
RAM : Random Access Memory
RC4 : Rivest Cipher 4
RFID : Radio Frequency Identification
RSA : Rivest-Shamir-Adleman
SOC : System on Chip
SPA : Simple Power Analysis
SPI : Serial Peripheral Interface
SRAM : Static Random Access Memory
SWD : Serial Wire Debug
UART : Universal Asynchronous Receiver Transmitter
USB : Universal Serial Bus

ÖZET

YÜKSEK LİSANS TEZİ

NESNELERİN İNTERNETİNDEKİ DONANIMLAR İÇİN YAN KANAL SALDIRILARINA KARŞI SİBER GÜVENLİK ANALİZİ

Mustafa ŞİRİN

İstanbul Üniversitesi-Cerrahpaşa

Lisansüstü Eğitim Enstitüsü

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Elektrik-Elektronik Mühendisliği Programı

Danışman : Prof. Dr. Fırat KAÇAR

Nesnelerin interneti cihazları içerdiği bilgilere göre saldırganların hedefi olabilmekte, siber saldırılara maruz kalabilmektedir. Bu sebeple nesnelerin interneti cihazlarının güvenlik değerlendirmeleri yapılmalı ve bulunan açıklar için gereken önlemler alınmalıdır. Nesnelerin interneti cihazlarına yapılan güvenlik değerlendirmeleri sadece yazılım ve haberleşme tarafında değil donanım tarafında da yapılmalıdır. Cihaza ait yazılım donanım üzerinde işlem yapıp çalışmasından dolayı kritik bilgi içeren donanımlarda cihazın donanımı, cihaz içerisinde bulunan veriler ve yapılan işlemler hakkında dışarıya istemsiz şekilde bilgi sızdırabilmekte ve saldırganlar bu sızıntıları analiz ederek cihaz veya üretici hakkında kritik bilgilere erişebilmektedir.

Şifreleme anahtarı, sistem parolası gibi kritik bilgiler içeren donanımların farklı yollardan dışarıya sızdırdığı bilgiler parola veya şifreleme anahtarı ile ilişkili olduğu durumda sızdırılan bilgiler kullanılarak sistem parolası veya şifreleme anahtarı elde edilebilir. Donanımların yaptığı istemsiz sızıntılar harcadığı güç, elektromanyetik yayılım, işlem süresi gibi farklı yollardan olabilir. Bu sızıntılara yönelik yan kanal saldırıları gerçekleştirilerek kritik bilgiler elde edilebilir. Yan kanal saldırıları tespit edilen sızıntıya ve analiz işlemine göre farklı şekillerde uygulanabilmektedir.

Yapılan tez çalışmasında iki farklı donanım belirlenmiş ve donanımların farklı senaryolar karşısında içerisindeki bilgileri dışarıya sızdırma durumları yan kanal saldırıları ile değerlendirilmiştir. Hazırlanan senaryolarda donanımdaki bilgi sızıntıları işlem süresi ve

harcanan güç olarak modellenmiştir. Bu sızıntıların değerlendirilmesi için zamanlama analizi ve basit güç analizi saldırıları denenmiştir. Yapılan deneylerde parolanın ve şifreleme anahtarının hangi senaryolarda güvende olduğu hangi senaryolarda ele geçirilebildiği sonuçları elde edilmiştir. Hazırlanan senaryolardan elde edilen sonuçlarla birlikte zamanlama analizi ve basit güç analizine dayanıklı sistem tasarımı konusuna katkı sağlamak amaçlanmıştır.

Aralık 2022 , 104 sayfa.

Anahtar kelimeler: Nesnelerin İnterneti, Donanım Güvenliği, Yan Kanal Saldırıları

ABSTRACT

M.Sc. THESIS

CYBER SECURITY ANALYSIS OF SIDE CHANNEL ATTACKS FOR INTERNET OF THINGS HARDWARES

Mustafa ŞİRİN

İstanbul University-Cerrahpaşa

Institute of Graduate Studies

Department of Electrical and Electronics Engineering

Electrical and Electronics Engineering Programme

Supervisor : Prof. Dr. Fırat KAÇAR

Internet of Things devices can be targeted by attackers according to the information they contain and can be exposed to cyber-attacks. For this reason, security evaluations of Internet of Things devices should be made and necessary countermeasures should be taken for the vulnerabilities found. Security evaluations of Internet of Things devices should be made not only on the software and communication side but also on the hardware side. Since the software of the device operates on the hardware, in the hardware that contains critical information, the hardware of the device can unintentionally leak information about the data contained and operations performed within the device, and attackers can access critical information about the device or the manufacturer by analyzing these leaks.

In cases where the information leaked by hardware containing critical information such as the encryption key and system password is related to the password or encryption key, the system password or encryption key can be obtained by using the leaked information. Unintentional leaks made by hardware can be in different ways such as power consumption, electromagnetic propagation, and execution time. Critical information can be obtained by performing side-channel attacks against these leaks. Side channel attacks can be applied in different ways depending on the leak detected and the analysis process.

In this thesis, two different hardware devices were determined and their situation to leak information in different scenarios was evaluated with side-channel attacks. In the scenarios prepared, the information leaks in the hardware are modeled as processing time and power

consumed. Timing analysis and simple power analysis attacks were tried to evaluate these leaks. In the experiments, the results were obtained in which scenarios the password and the encryption key were secure and in which scenarios they could be seized. With the results obtained from the prepared scenarios, it is aimed to contribute to the issue of system design resistant to timing analysis and simple power analysis.

December 2022, 104 pages.

Keywords: Internet of Things, Hardware Security, Side Channel Attacks

1. GİRİŞ

İnternetin farklı ürün segmentlerinde yaygınlaşması ve internet üzerinden aktarılan bilginin çeşitliliğinin artması beraberinde aktarılan bilgilerin ne kadar güvenli aktarıldığı, 3. kişilerin bilgilere erişme durumu, aktarılan bilginin doğru ve değiştirilmemiş bilgi olması gibi sorunları getirmiştir. Özellikle kişisel bilgilerin, bir uygulamaya ait hesap bilgilerinin veya ödeme sistemlerindeki bilgilerin güvenliğini sağlamak giderek önem kazanmıştır. Saldırganlar artan bilgi paylaşımı ve bilginin değeri düşünüldüğünde çok farklı senaryolarda sistemlere saldırmakta ve zarar vermektedir. Günümüzde kullanımı giderek yaygınlaşan bir kavram olan ve birçok farklı sektörde farklı ürünlerde kullanılan Nesnelerin İnterneti (Internet of Things - IoT) teknolojisi ile uzaktan kontrol edilebilen cihazlar saldırganların hedefi haline gelmiştir. Kullanım yerine göre küçük boyutlarda ve düşük kapasiteli işlem gücüne sahip olabilen bu akıllı cihazlar saldırganlar tarafından manipüle edilebilmekte ve kullanıcısının bilgilerini saldırganlara iletebilmektedir.

IoT cihazlarına yönelik yapılan birçok saldırı çeşidi bulunmaktadır. Cihazın haberleşme yapısına saldırı yapılabileceği gibi içerisindeki yazılım ele geçirilip değiştirilebilmekte, cihazın yanlış bilgiler üretmesi sağlanarak hem cihazın kullanıcısına hem de kullanıldığı ortama zarar verilebilmektedir. Sadece kullanıcının bilgisi değil aynı zamanda üretici firmanın üretim bilgisi cihaz üzerinde tersine mühendislik saldırısı ile ele geçirilebilmektedir. Cihaz üzerinde birçok saldırı yazılım ve haberleşme odaklı yapılsa da cihazın donanımının güvenliğini sağlamak da çok önemlidir. Özellikle şifreleme yapan, içerisinde çok kritik bilgiler barından cihazların donanımları saldırganlar tarafından ele geçirildiğinde çok ciddi zararlara neden olabilmektedir. Donanım tarafında bulunan güvenlik açıklarının sisteme olan zarar verici etkisi çok fazla olmaktadır. Donanım güvenliği kapsamında donanımın birçok farklı noktasına yapılabilecek farklı saldırı çeşitleri bulunmaktadır. Bu saldırı çeşitleri beraberinde birçok güvenlik açığını da getirmektedir.

Donanım güvenliği içerisinde bir saldırı çeşidi olan Yan Kanal Saldırıları konusunda hazırlanan bu tez çalışmasında içerisinde parola bulunan ve doğru parola girilmeden sistem erişiminin sağlanmadığı bir sistem iki farklı mikrodenetleyici üzerinde kurulmuş ve sistem birçok farklı parola doğrulama senaryosunda kurgulanmıştır. Kurgulanan senaryolarda sistem içerisinde

doğrulama yapan parolayı elde etmek için yan kanal saldırı çeşitlerinden birisi olan zamanlama analizi saldırısı gerçekleştirilerek farklı senaryoların güvenlik durumları değerlendirilmiştir. Ayrıca bir mikrodenetleyici üzerinde bir şifreleme algoritması çalıştırılmış ve mikrodenetleyici üzerinde yan kanal saldırı çeşitlerinden biri olan basit güç analizi saldırısı gerçekleştirilmiştir. Şifreleme anahtarının elde edilmesinin amaçlandığı uygulama farklı senaryo kurguları denenerek sistemin güvenlik durumları değerlendirilmiştir.

Hazırlanan tez çalışmasında, ilk olarak nesnelerin interneti kavramı anlatılmış ve güvenlik durumları incelenmiştir. Ardından donanım güvenliği, saldırı çeşitleri ayrıca bilgi güvenliği için büyük öneme sahip olan kriptografi konusu anlatılmıştır. Tez çalışması kapsamında yapılan donanım saldırılarından olan yan kanal saldırıları genel olarak anlatılmış, güç analizi ve zamanlama analizi açıklanmıştır. Ardından tez çalışması için hazırlanan deney test ortamı sunulmuş ve testler sonucu elde edilen bulgular paylaşılmıştır. Bulgular ışığında elde edilen sonuçlar değerlendirilmiş ve güvenlik konusunda önerilerde bulunulmuştur.

2. KAVRAMSAL ÇERÇEVE

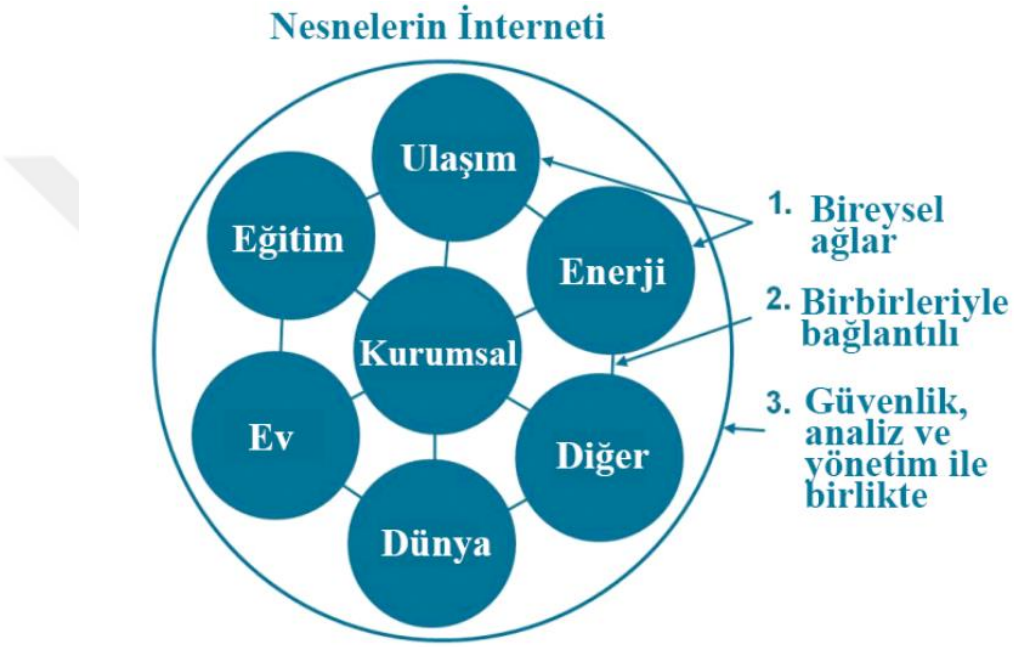
2.1. NESNELERİN İNTERNETİ VE GÜVENLİĞİ

Teknolojinin hem fiyat olarak ulaşılabilir olması hem de üretici sayısındaki artış ile akıllı cihazların gündelik hayatımızın bir parçası haline gelmesi sonucunda "Nesnelerin İnterneti" kavramı sıklıkla karşımıza çıkmaktadır.

Nesnelerin İnterneti terimi ilk olarak Forbes dergisinde yayınlanan bir yazıyla teknoloji dünyasının karşısına çıkmıştır [1]. Massachusetts Teknoloji Enstitüsü'nde (MIT) şirketler arası RFID (Radio Frequency Identification - Radyo Frekansı ile Tanımlama) altyapısı tasarlanmaya başlamasıyla popülerlik kazanan Nesnelerin İnterneti, MIT'de çalışan Kevin Ashton'un 2002 yılında 'Nesneler için bir internete (İngilizcesi: an internet for things), bilgisayarların gerçek dünyayı anlayabilmesi için standartlaşmış bir yola ihtiyacımız var' cümlesinin alıntılandığı Forbes makalesinin [2] "The internet of things" başlığıyla kavram olarak hayatımıza girmiştir. Kevin Ashton 2009 yılındaki bir yazısında ise Nesnelerin İnterneti terimini ilk olarak 1999 yılında Procter & Gamble'da (P&G) yaptığı bir sunumda bu terimin yöneticilerin dikkatini çekeceğini düşünerek kullandığını belirtmiştir [3].

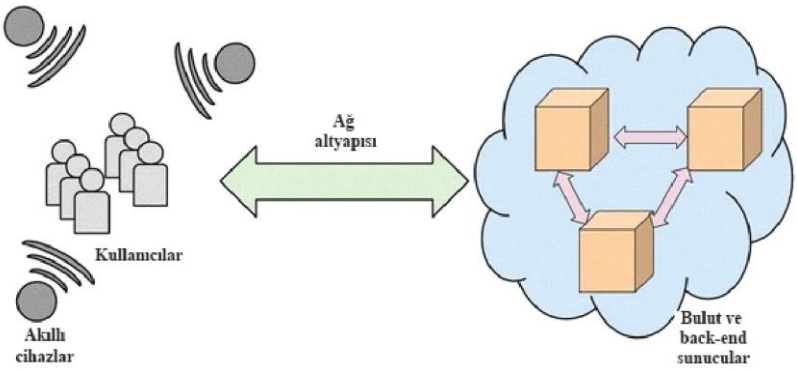
Literatürde Nesnelerin İnterneti'nin kavram olarak birçok tanımlaması yapılmıştır. Bunlardan bir tanesi olan [4]'te "Sosyal, çevresel ve kullanıcı bağlamlarında bağlantı kurmak ve haberleşme yapmak için akıllı arayüzler kullanan akıllı alanlarda çalışan, kimlikleri ve sanal kişilikleri olan şeyler" olarak yapılan tanımlamadır. Perera ve diğ. [5] Nesnelerin İnterneti'ni, internete bağlı nesnelerden, geleceğin internetinin bir parçası olan akıllı cihazlar ve diğer araçlardan oluşan dinamik küresel bir bilgi ağı olarak tanımlamıştır. Sensörlerden akıllı telefonlara, buzdolaplarından akıllı saatlere ve robot süpürgelere kadar uzanan birbirine bağlı nesnelerin ağı olan Nesnelerin İnterneti bu nesnelerin entegrasyonunu sağlamaktadır. Nesnelerin İnterneti teknolojisinin amacı nesnelerin birlikte çalışabileceği haberleşme protokolleri ve yazılımlar ile bir ağ altyapısı sağlamaktır [6]. Sağlanan entegrasyon ile nesneler ağda birbirlerine bağlanarak yaptıkları işlemlerin sonuçlarını bir başka nesneye gönderebilir, bir başka nesneden ihtiyaç duyduğu bilgiyi alabilir duruma gelmektedir. İnternet üzerinden haberleşme ile insanlar bu nesnelerden bilgi alabilmekte ve bu nesnelere, nesnelerin yakınında

olmasalar dahi, komut gönderebilmektedir. Nesnelerin yapısal farklılıkları ve sayılarıyla birlikte IoT nesnelerin haberleşmesi amacıyla oluşturulan bir ağ koleksiyonundan oluşur. Örneğin binalardaki ısıtma, havalandırma, telefon, güvenlik ve aydınlatma gibi çeşitli sistemler bulunmaktadır. IoT ile bu sistemlerin ortak bir ağda haberleşerek iş birliği yapacağını söyleyebiliriz [7]. Bununla birlikte bu sistemlerin hem verimliliği artacak hem de kontrolü kolayca sağlanabilecektir. Bina örneğini çok daha bir büyük bir ekosisteme uyarladığımızda Nesnelerin İnternetinin ağların ağı olarak gösterimi Şekil 2.1'de verilmiştir.



Şekil 2.1: IoT yapısı ağların ağı şeklinde görülebilir [7].

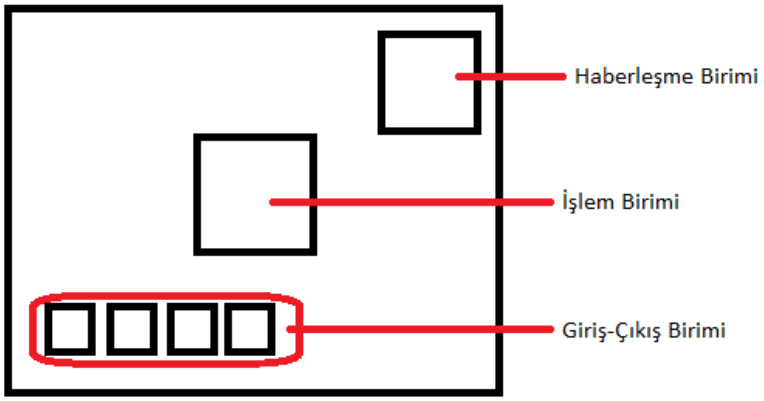
Lopez ve diğ. [8] çalışmasında, basit bir IoT sisteminin yapısını Şekil 2.2'de gösterilen akıllı cihazlar (nesneler), ağ altyapısı ve arka uç sunucular olmak üzere üç ana bileşenden oluşturmuştur. Bu basitleştirilmiş mimari, IoT teknolojisinin arkasındaki özü tanımlamaktadır. Şekil 2.2'de görülen akıllı cihazlar hem kullanıcılarla hem de ağa bağlı diğer cihazlarla etkileşime girecek şekilde tasarlanmıştır hatta bu cihazlardan bazıları doğrudan kullanıcılarla etkileşime bile gerek duymayabilir [8].



Şekil 2.2: IoT'nin basitleştirilmiş yapısı [8].

Nesnelerin İnterneti'ndeki nesneler geniş bir kümeyi kapsamaktadır. Literatürde bu nesnelerin kimisi akıllı cihazlar, mobil cihazlar, akıllı nesneler gibi farklı isimlendirilebilmektedir. Stojkoska ve diğ. [9] çalışmalarında akıllı cihazları, basit sensör düğümlerinden ev aletlerine ve akıllı telefonlara kadar hesaplama ve haberleşme yapabilen nesneler olarak değerlendirmiştir [9]. HaddadPajouh ve diğ. [10] çalışmasında nesnelerin insan etkileşimi ile ya da insan etkileşimi olamadan haberleşme yapabilen ve işlemler gerçekleştiren bir nesneyi ifade ettiğini, bunların sensörler, otomobiller, buzdolapları, bulaşık makineleri, televizyonlar ve daha pek çok şey olabileceğini belirtmiştir [10]. Özellikle gündelik hayatımızda kullandığımız ev eşyalarında "akıllı" ön ekiyle kullanılan cihazlar bu ön eki, eski versiyonlarında olmayan bilgiyi işleyebilme, hesaplama yapma, interneti kullanabilme ve internet üzerinden kullanılabilme özellikleri sayesinde taşımaktadır.

Nesnelerin İnterneti cihazlarının yapısını incelediğimizde temelde internete bağlanabilen ve kendisine yönlendirilen işi yapıp sonucunu ileten bir sistem olarak karşımıza çıkmaktadır. Nesnelerin, internete bağlanabilen ve içerisinde hesaplama yapıp sonuç çıkartabilen aygıtlar olduğu göz önüne alındığında basit şekilde bir nesnenin Şekil 2.3'te gösterildiği gibi bir donanım ve donanım içerisinde işlem birimi, haberleşme birimi ve giriş çıkış birimi içerdiği söylenebilir.



Şekil 2.3: Basit şekilde bir nesnenin yapısı.

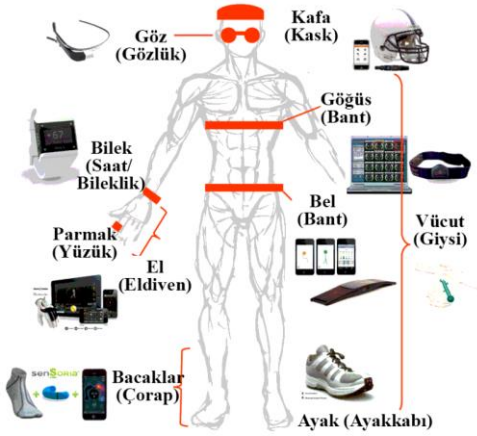
Nesnenin geniş kapsamlı tanımı ve internetin artık birçok yerde kolayca erişilebilir olması ile akıllı cihazlar yalnızca evlerde değil fabrikalarda, arabalarda, parklarda ve daha birçok alanda kullanılmaktadır.

Song ve diğ. [11] çalışmasında IoT teknolojisinin akıllı lojistik geliştirmenin anahtar teknolojilerinden biri olduğunu, bir akıllı lojistik sisteminin, IoT tabanlı çeşitli sensörler, RFID etiketler gibi araçlar kullanarak kargo, nakliye rotası ve depolama gibi bilgilerin takibini sağlayabildiğini belirtmiştir. İlgili çalışmada IoT'nin temel fikrinin etkileşime giren çeşitli nesnelerin iş birliği yaparak insan müdahalesi olmadan bir bilgiyi anlamlandırma ve hedefine ulaştırma olduğu bahsedilmiştir [11]. Haberleşme sadece insandan nesneye değil nesneden nesneye yapılabilmektedir. Fabrikaların üretim hatlarında akıllı sensörler aracılığıyla yapılan ölçümlerin sonuçlarının üretim hattındaki diğer makinelere iletilmesi ve hat üzerindeki makinelerin ilgili değerlere göre işlem yapması ve ardından bir sonraki aşamadaki sisteme durum bilgisini vermesi ile tüm bu sürecin insan müdahalesi olmadan yapılabileceğini göstermektedir. Böylelikle üretim hattı çalışanı fabrikaya gitmeden, uzaktan, sistemlerin ve üretimin durumunu kontrol edebilir hale gelmektedir. Sensörlerin düşük güçte çalışabilmesi, boyutlarının küçük olması gibi özellikleri ile birçok sektörde kullanılarak uzaktan takip edilen sistemlerin geliştirilmesi sağlanabilir. Ulaşım sektöründe bir yolcunun otobüsünü anlık olarak izleyebilmesi, mobil ödeme sisteminin otobüslere entegre edilmesi, trafiğin yoğunluğunun anlık izlenebilmesi ve buna bağlı olarak en az yoğunluktaki bölgeden rota planlaması yapılabilmesi gibi birçok sistem IoT teknolojilerinin ulaşım sektöründe kullanılmasına örnek olarak verilebilir. Bunların yanında evlerde kullandığımız akıllı televizyonlar, akıllı kahve makineleri, akıllı buzdolapları gibi cihazlar ayrıca giyilebilir teknolojilerden olan akıllı saatler,

marketlerde kullandığımız elektronik ödeme cihazları yaygın olarak kullanılan IoT teknolojilerine yine örnek olarak verilebilir.

2019 yılı sonu itibariyle insanların hayatını global ölçekte etkileyen, temas yoluyla bulaştığı için yüz yüze görüşmeleri tehlikeli kılan Covid-19 pandemisi ile uzaktan ve evden çalışma kavramı popülerlik kazanmış, insanlar internet üzerinden bağlantıda kalarak işlerini, eğitimlerini devam ettirmeye çalışmışlardır. Artık günlük hayatımızın bir parçası haline gelen akıllı telefonlar, akıllı saatler ile giderek artan IoT cihaz kullanımı sadece tüketici sektöründe değil eğitim, otomasyon teknolojileri, ödeme terminalleri, güvenlik sistemleri, yangın alarmları gibi birçok farklı alanda üretim, ulaşım, sağlık, finans gibi birçok sektörde kullanılmakta ve her geçen gün popülerliği ve kullanımı artmaktadır. 2022 yılı Eylül ayında Statista platformunda yayınlanan IoT pazar değeri araştırmasına göre 2020 yılında yaklaşık 389 milyar Amerikan doları olan IoT pazarının 2030 yılında 1 trilyon Amerikan dolarının üzerinde olacağı beklenmektedir. Giderek artan akıllı cihaz kullanımı ile birlikte IoT teknolojisinin kullanıldığı tüm pazarlar içerisinde tüketici pazarının gelecek 10 yılda yüzde 35'lerden yüzde 45 civarı kullanıma çıkacağını belirten raporda aynı zamanda IoT bağlantılı cihaz sayısının da bu süre zarfında 3 katına çıkacağı tahmini de yer almaktadır [12]. 2020 yılında 9,7 milyar adet IoT cihazının olduğunu belirten Dünya Geneli IoT Bağlı Cihazlar başlıklı Statista raporunda ise 2030 yılında bu sayının 29 milyara çıkacağı tahmin edilmektedir [13].

İnternetin yaygınlaşması ve bununla birlikte önceki bölümlerde anlatıldığı gibi internete bağlı cihaz kullanımının artması, IoT cihazlarının adeta etrafımızı sarması dolasıyla bu nesnelerin taşıdığı bilgilerin önemi artmaktadır. Nesnelerin topladığı, işlediği, sakladığı veya bir sonraki düğüme ilettiği bilgiler saldırganların hedefi olabilir. Bu durum IoT kullanıcısının veya üreticisinin önemli bilgilerinin saldırganlar tarafından çalınmasına da neden olabilir. Örneğin son kullanıcıların günümüzde sıklıkla tercih ettiği giyilebilir cihazlar işlevsellik olarak çeşitlilik göstermektedir. Baş, gözler, bilek, bel, eller, parmaklar, bacaklar gibi vücudun çeşitli kısımlarına takılmanın yanı sıra farklı kıyafetlere de gömülü olarak yerleştirilerek kullanılabilmektedir. Şekil 2.4'te örnek olarak kullanılabilecek IoT giyilebilir cihazları gösterilmiştir [5].

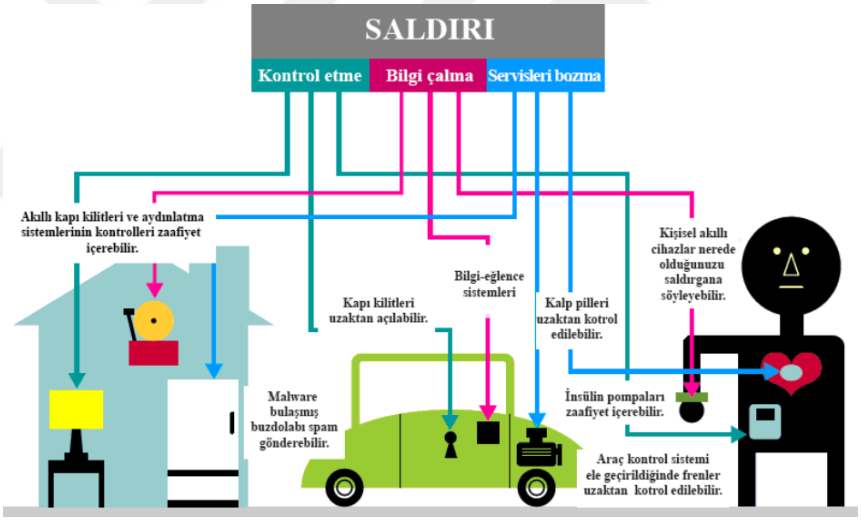


Şekil 2.4: Endüstri pazarında giyilebilir IoT çözümleri tarafından popüler olarak hedeflenen farklı vücut bölümleri [5].

Şekil 2.4'teki örnekte gösterilen cihazlar kullanıcının sağlık verilerini toplayabilmekte, konumunu görebilmekte, günlük rutinlerini izleyebilmektedir. Kısacası kullanıcının kişisel sağlık verileri bu cihazlar içerisinde işlenmekte, iletilmekte veya kaydedilmektedir. Bu durum kişisel bilgilerin güvenliğinin sağlanmasını gerektirmekte ve cihazların kullanıcı izni dışında farklı yer veya kişilere bu bilgileri sızdırmaması gerekmektedir. Sadece giyilebilir teknolojiler değil internete bağlı nesnelerin, akıllı şeylerin etrafımızı sarması, sensör teknolojilerinin IoT sistemleri ile entegre çalışması bireysel mahremiyet konusunda önemli bir tehdit oluşturabilmektedir. Kullanıcılar, bilgilerini tehlikeye atmalarına yol açan internette yaptıkları hatalar veya işlemlerin aksine artan sayıda algılama cihazı ile şimdiye kadar hiç beklenmedik durumlarda farkına bile varmadan veri toplama hedefleri haline gelmektedir [8].

Son yıllarda IoT cihazlarına yapılan siber saldırıların, elektrikli şarj istasyonlarına sızmaktan güvenlik kameralarına ve daha fazlasına kadar genişlediği görülmektedir. Nisan 2022'de yayınlanan bir haberde saldırganlar elektrikli şarj istasyonlarını hedef alarak otoparkın web sayfasını ele geçirdiği bilgisi yer aldı [14]. BBC'de yer alan bir haberde ise çocukların konum takibi için geliştirilen bir akıllı saatin kolayca hacklenebileceği, saldırganların cihazları ele geçirdikten sonra elde edeceği bilgilerin neler olduğu bilgisi paylaşıldı [15]. Yine BBC'de yayınlanan bir haberde bir üreticinin kablosuz kameralarının ele geçirilebileceği bilgisi paylaşıldı. Haberde yer alan bu güvenlik açığıyla birlikte test ekibi uzaktaki bir kameradan görüntü alabildiklerini açıkladılar [16]. Eylül 2022 ayında yayınlanan bir haberde ise İngiltere'nin en büyük toplu taşıma firmalarından birine yönelik siber saldırı yapıldığı bilgisi paylaşıldı [17]. IoT ürünlerindeki güvenlik riskleri sadece yapılan siber saldırılar sonrası değil üretim sonrası ya da kullanım aşamasında, test firmaları aracılığı ile ortaya çıkarılabiliyor.

Böylelikle kötü niyetli bir etkinlik olmadan üretici firmaya bildirilen güvenlik açıkları firma tarafından hızlıca kapatılıyor. FutureIoT platformunda yayınlanan bir haberde bir tehdit istihbarat ekibinin bulduğu 10 farklı firmanın cihazlarını etkileyen 56 güvenlik açığı bilgisi yer aldı. Araştırmacılar buldukları güvenlik açıklarını: güvenli olmayan protokoller, zayıf şifreleme veya hatalı kimlik doğrulama şemaları, güvenli olmayan ürün yazılımı güncellemeleri ve uzaktan kod çalıştırma olarak 4 ana kategoride hazırladıkları raporda sunmuşlardır [18][19]. Artan kullanım ile gelen cihaz sayısındaki artış ve üretilen cihazların güvenlik açıklarının bulunmasıyla saldırılar da artmaktadır. Statista'da yayınlanan 'Küresel IoT Siber Saldırı Sayısı 2020-2021' başlıklı raporda Ekim 2020'de dünya genelindeki IoT saldırılarının sayısının 10,8 milyonu aştığı, 2021 yılında aynı ayda yaklaşık 6 milyon saldırı olduğu tespit edilmiştir [20]. Saldırı sayısında bir önceki yıla göre bir azalma olmasına karşın 6 milyon saldırı çok büyük bir cihaza etki edebilecek ve çok önemli bilgiler ele geçirebilecek etkiyi göstermektedir. Burada üreticilerin bilinen açıkları kapatması da bu saldırıların düşmesine neden olmuş olabilir.



Şekil 2.5: Çevremizde siber saldırıya uğrayabilecek cihazlar [21].

IoT cihazlarına yönelik saldırılarda amaç sadece cihaz içerisindeki bilginin deşifre edilmesi, iletişimin dinlenmesi veya kullanıcıya ait özel bilgilere erişilmesi değil aynı zamanda cihazın ele geçirilerek zombi bilgisayar yani saldırgan tarafından ele geçirilmiş virüslü makine olarak kullanılabilmesi de olmaktadır. Saldırganlar IoT cihazları ele geçirerek birden fazla zombi cihazla botnet ağı kurabilir ve bu botnet ağıyla güçlü bir DDoS (Distributed Denial-of-Service – Dağıtılmış Hizmet Reddi) saldırısına neden olabilir. Bu durumun bir örneği 2016 yılında ComputerWorld'de yer alan bir haberde paylaşılmıştır. İlgili haberde DNS (Domain Name System – Alan Adı Sistemi) sağlayıcısına yönelik yapılan saldırının internete bağlı kahve

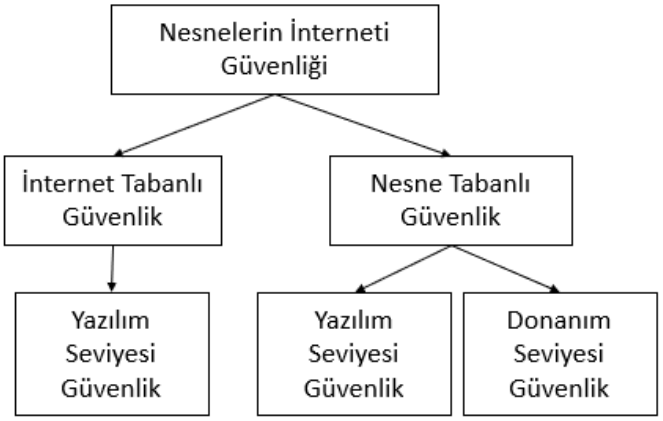
makineleri, çamaşır makineleri ve buzdolapları gibi cihazlarla yapıldığı bilgisi yer almaktadır [22]. Kahve makinesi, televizyon veya çamaşır makinesi gibi akıllı cihazlar bilgisayarlara nazaran çok daha az işlem gücüne sahip olmasına karşın botnet ağına dahil edilen yüz binlerce akıllı cihazla çok büyük bir saldırı gücü elde edilebilmektedir. Bu konuda Bertino ve diğ. [23] yayınladığı makalede IoT cihazlarının DDoS konusundaki güvenlik risklerinden ve korunma tekniklerinden bahsetmişlerdir ve şifreleme, kimlik doğrulama, erişim kontrolü, ağ güvenliği ve uygulama güvenliği gibi mevcut güvenlik mekanizmalarını IoT ekosistemine uyacak şekilde genişletmenin çok önemli olduğunu belirtmişlerdir [23].

OWASP'ın (Open Web Application Security Project) IoT cihazların siber güvenlik açıkları konusunda 2018'de yayınladığı Internet of Things Top 10 2018 listesinde IoT cihazlar için en önemli 10 güvenlik açığı listelenmiştir [24]. Bunlar:

- Zayıf, tahmin edilebilir veya sabit ayarlanmış parolalar
- Güvensiz ağ hizmetleri
- Güvensiz ekosistem arayüzleri
- Güvenli güncelleme mekanizmasının eksikliği
- Güvensiz veya eski bileşenlerin kullanımı
- Yetersiz gizlilik koruması
- Güvensiz veri aktarımı ve depolama
- Cihaz yönetimi eksikliği
- Güvensiz varsayılan ayarlar
- Fiziksel güvenlik eksikliği

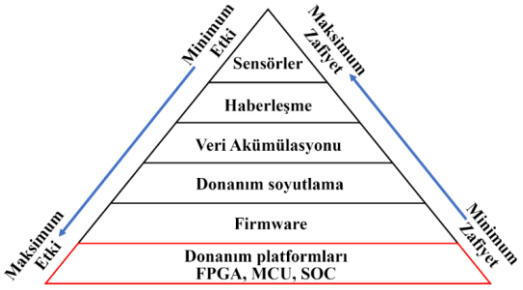
şeklindedir [24]. Bunun yanında OWASP IoT güvenlik açıkları tablosu oluşturarak güvenlik açıklarını ve saldırı yüzeylerini tablo haline sunmuştur. Bu tabloda zayıf parolalar, 2 faktörlü doğrulama, hizmet reddi, şifrelenmemiş servisler, hafızanın çıkartılması gibi birçok güvenlik açığı ve saldırı yüzeyleri listelenmiştir [25].

Piyasaya sürülen ve internete bağlanabilen sistemlerin kullanıcıların gizliliklerini koruyabilecek şekilde güvenliğini sağlaması beklenmektedir. Özellikle sağlık sektöründe ve savunma sanayinde kullanılan cihazların içerdiği kritik bilgilerin dışarıya sızmaması gerekmektedir. Bunun için üretilen cihazların kapsamlı bir güvenlik sıkılaştırması yapılması ve saldırılara karşı korunması gerekmektedir. IoT cihazlarının boyut olarak küçük olabilmesi bu nesnelere ekstra güvenlik modülü takılmasını zorlaştırmaktadır. Aynı şekilde düşük işlem kapasiteleri sebebiyle karmaşık güvenlik algoritmaları için uygun değillerdir [26]. [27]'de IoT'deki güvenlik sorunları hizmet reddi saldırıları, tekrarlama saldırıları, parola tahmin etme saldırıları, kimlik sahtekarlığı ve içeriden yetkili saldırı olarak listelenmiş ve detaylandırılmıştır. Bunların yanında fiziksel olarak cihazın güvenliği de gittikçe önem kazanmakta ve donanım düzeyindeki tehditler artmaktadır. [26]'da IoT güvenliği yazılım ve donanım olarak iki ana bölüme ayrılmış ve donanım düzeyindeki güvenlik sorunları incelenerek donanım truva atları anlatılmıştır.



Şekil 2.6: Nesnelerin İnternetinde Güvenlik [26].

Bir IoT cihazının farklı bileşenlerinde farklı zafiyetler ve bu zafiyetlerin farklı boyutlarda etkileri olabilir. Örneğin nesnenin haberleşme tarafında iletişim dinlenebilirken donanım tarafında içerisinde bulunan bilgiler ele geçirilebilir. Şekil 2.7'de nesnenin bileşenlerinin güvenlik açıkları ve etkilerine göre oluşturulmuş bir piramit gösterilmektedir [28].



Şekil 2.7: IoT cihazlarında güvenlik açıkları ve etkileri piramidi [28].

Şekil 2.7'den görüleceği üzere donanım tarafında yer alan güvenlik açıklarının etkileri büyük olmaktadır. Bunun nedeni donanım seviyesinde bir mikroişlemcide yer alan bir güvenlik açığı yazılımsal olarak düzeltilemeyebilir ve değiştirilmesini gerektirebilir. Donanım tarafında istemsiz bilgi sızıntıları bilgi çalınmasına neden olabilir. Bu durum Bölüm 3.1'de anlatılan Yan Kanal Saldırıları ile meydana gelmektedir.

Nesne içerisindeki bilginin donanım içerisinde saklanması ve işlenmesi nesne donanımını siber hedef haline getirebilmektedir. Donanım tarafında keşfedilen bir açık o donanımı kullanan diğer cihazları da etkileyeceği için ilgili nesnenin ürün grubunun zafiyetli olmasına neden olabilir. Bu yüzden IoT cihazlarda yazılım güvenliğinin yanında donanım güvenliğini de sağlamak çok önemlidir.

2.2. DONANIM GÜVENLİĞİ

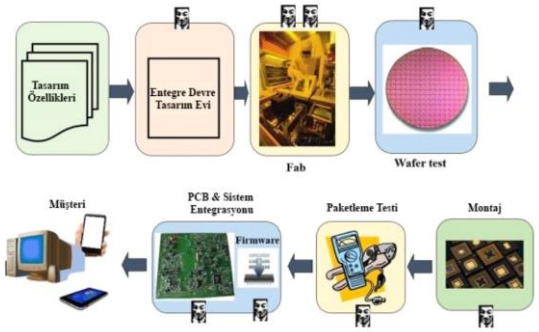
Bir nesneyi internete bağlayarak uzaktan kullanabilmek kullanıcı konforu dolayısıyla sıklıkla tercih edildiğinden artan internete bağlı cihaz sayısıyla güvenlik sadece internet haberleşmesi veya yazılımda değil donanım için de çok önemli bir konu haline gelmiştir. IoT cihazlarına yönelik her geçen gün artan saldırı türleri, donanımı kullanarak ürün içerisindeki bilgiye veya yazılıma erişmeyi de içermeye başlamıştır.

Genellikle IoT cihazlarının donanımını oluşturan araçlar, karmaşık sistemler içerisinde belirli işlevleri yerine getiren özel bilgisayar olarak açıklayabileceğimiz gömülü sistemler adı verilen sistemlerdir. Bir gömülü sistemde işlevleri kısıtlamak odak bir iş yapılması için küçük bir alan, hızlı işlemler ve verimliliği beraberinde getirmektedir [21]. Örneğin bir akıllı buzdolabının donanımı yaptığı işlemlerle buzdolabının çalışmasını ve internet üzerinden kontrol edilmesini sağlar, yine aynı şekilde akıllı su sayacı ev kullanıcısının tükettiği su miktarını ölçümler ve internet üzerinden bu bilgiyi faturasına yansıtır. Bu araçlar bu kapsamda üretilerek bu işler için

uygun hale getirilirler, yani bir akıllı su sayacı ile web üzerinde gezinme yapılamamaktadır. Öte yandan IoT cihazlarının sınırlı işlem kapasitesi, düşük güç tüketimi gibi kısıtlamaları dolayısıyla üretilen birçok üründe şifreleme gibi güvelik önlemleri alınmamaktadır. Bu nedenle bir akıllı su sayacı ile web sayfalarında gezinemezken yüz binlercesi bir botnet ağına dahil edilerek büyük bir siber saldırı gerçekleştirilebilir. İnternete bağlı bir cihazın, kullanıcı bilgileri, ağ bilgileri gibi önemli bilgi içerdiği ve taşıdığı için kapsamlı bir güvenlik sağlaması gerekmektedir. Bu kapsamlı güvenlik hem yazılım tarafında hem network tarafında hem de donanım tarafında sağlanmalıdır.

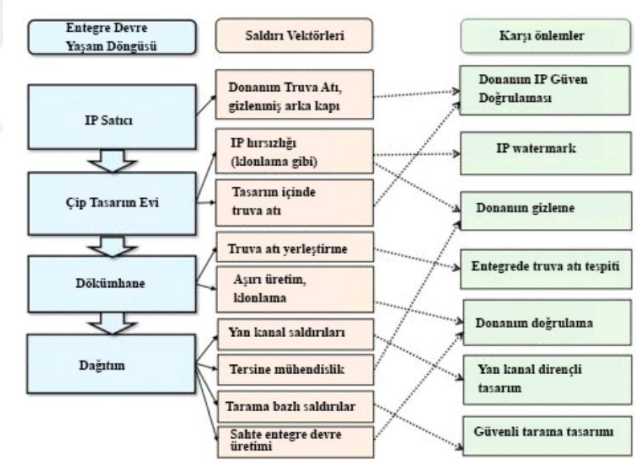
Donanım saldırıları konusunda literatür incelendiğinde oldukça fazla sayıda saldırı çeşidi ve tehdit ile karşılaşılmaktadır. Donanım güvenliği tarafındaki çalışmalarda sadece üretilen bir entegreye fiziksel olarak yapılan saldırılar değil ayrıca üretim aşamasında çip tasarımına yapılan saldırılar da gündeme gelmiştir. Çipin IP (Intellectual Property) bilgisi, filigranı gibi tasarımsal bilgiler üretim hattının güvenirliğine bağlı olarak tehlike altında kalabilmektedir. Bu aşamada yapılan bir saldırı ile çip üzerine Donanım Truva Atı eklenebilmekte veya tasarım bilgisi kopyalanabilmektedir. Bu durum üretilen çipin kontrol aşamalarında fark edilmemesiyle birlikte büyük sorunlara ve kayıplara neden olabilmektedir. Bunların dışında: çipe ait güç veya elektromanyetik sinyal, zamanlama sızıntısı gibi bilgilerin çip sinyallerinin ölçülerek ve analiz edilerek elde edilmesiyle meydana gelen yan kanal saldırıları, tersine mühendislik ile çip tasarımının ele geçirilmesi, baskı devre kartındaki (PCB - Printed Circuit Board) devre yollarının ve komponentlerin kurcalanması, PCB üzerindeki veri yollarının dinlenmesi, test/hata ayıklama mekanizması yoluyla gömülü sistem üzerindeki kaynaklara ayrıcalıklı erişim gibi saldırılar donanım saldırılarına örnek verilebilir. Donanımın hem üretim aşaması hem de üretim sonrası güvenlik açısından önemli olduğundan bu noktada karşımıza Donanım Güveni (İng. Hardware Trust) kavramı çıkmaktadır. Donanım güveni bir donanımın yaşam döngüsüne güvensiz varlıkların dahil edilmesinden kaynaklanmaktadır. Bu güvensiz varlıklar bir donanım bileşenin veya sistemin güvenilirliğini tehlikeye atabilmektedir. Güven sorunları genellikle güvenlikle ilgili endişelere yol açmaktadır [29]. Bir çip üretilirken birden fazla aşamadan geçtiği ve bu aşamalarda birçok farklı firma arasında dolaşabildiği için donanım güveni sorunları ortaya çıkabilmektedir. Konuya bu kapsamda bakıldığında güvenliğin, üretim sürecinin güvenilirliğinden başlayarak ele alınması gerektiği ve tasarım aşamasında güvenlik açıklarına dikkat edilmesi, üretim sonrası açıkların tespit edilip hızla kapatılması gerektiği

görülmektedir. Bir donanımın tasarımından kullanıcının eline geçene kadar olan geçtiği süreç Şekil 2.8'de gösterilmiştir.



Şekil 2.8: Donanım tasarım ve test süreci akışındaki adımlar ve sızıntı olabilecek noktalar [29].

Şekil 2.8'de görüleceği gibi saldırılar donanımın tasarım evresinde başlayıp her aşamada olabilmektedir. Şekil 2.9'da bir entegrenin yaşam döngüsü içerisinde karşılaşabileceği saldırı vektörleri ve alınabilecek karşı önlemler gösterilmektedir.



Şekil 2.9: Bir çipin yaşam döngüsündeki her aşama için saldırı vektörleri ve karşı önlemleri [29].

Donanım güvenliği yönelik farklı saldırı çeşitleri literatürde farklı kategorilere ayrılmıştır. [29]'da yapılan çalışmada saldırı yüzeyleri belirtilmiş ve örneklendirilmiştir. [30]'da farklı bir saldırı sınıflandırılması yapılmış ve saldırılar ilgili sınıflara ayrılmıştır. [31]'de yapılan saldırılar donanım güvenlik tehditleri olarak kategorilere ayrılmıştır.

[29]'da donanım güvenliği kapsamında saldırı yüzeyleri donanım saldırılarının başladığı düzeye göre sınıflandırılmıştır. Yapılan çalışmada saldırı yüzeyini olabildiğince küçük tutmanın karşı önlemler geliştirme konusunda ortak bir hedef olduğuna dikkat çekilmiştir.

Saldırı yüzeyleri: Çip Seviyesi Saldırıları, PCB Düzeyinde Saldırılar ve Sistem Düzeyinde Saldırılar olmak üzere 3 ana sınıfa ayrılmıştır. Çip düzeyi saldırılar doğrudan çiplere yönelik yönelik yapılan ve tersine mühendislik, klonlama, truva atı yerleştirme gibi saldırıları içeren saldırı yüzeyini tanımlar. Bu kapsamda yapılan saldırıdan orijinal çipin kopyası oluşturulabilir, çipe donanım truva atı ekleyerek uzaktan kontrol sağlayabilir veya arıza tehdidi oluşturulabilir. PCB düzeyi saldırılarda PCB'de tersine mühendislik yapmak, kartı yeniden tasarlamak ve sahte birimler oluşturabilmek için kartın şemasını elde etmek amaçlanmaktadır. PCB'ler tersine mühendislik ve kurcalama işlemleri konusunda çiplerden daha kolay bir hedef olduğundan basit bir optik inceleme veya sinyal işleme ile devre tasarımı elde edilebilir. Sistem düzeyi saldırılarda saldırganlar PCB üzerindeki açık kalmış haberleşme ve debug yollarını kullanarak veya bellek modülüne erişerek sistem üzerinde yetkisiz kontrol ve hassas bilgilere erişim sağlayabilir.

[30]'da donanım güvenliği kapsamındaki saldırı yöntemleri invaziv, yarı invaziv ve invaziv olmayan olarak 3 ana kategoriye ayrılmıştır. İnvaziv saldırılarda çip içerisine doğrudan erişim sağlanarak içerisinden bilgi çıkarmak ve işlevlerini anlamak amaçlanır. Bu tip saldırıda çip zarar görebilir. Yarı invaziv saldırılarda çip yüzeyi üzerinden çip içerisindeki yollara zarar vermeden bilgi edinmek amaçlanır. Çipin iç yapısı zarar görmez. İnvaziv olmayan saldırılarda ise çipin veya cihazın arayüzleri (gerilim, clock vb.) kullanılarak çip veya cihazla etkileşim kurmak amaçlanır. Fiziksel bir zarar oluşturmaz. Tersine mühendislik, mikroproblama invaziv saldırılara, optik görüntüleme yarı invaziv saldırılara, yan kanal saldırıları invaziv olmayan saldırılara örnek olarak verilebilir.

[31]'de donanım güvenlik tehditleri: mimari ve sistem tehditleri, gizli ve yan kanallar, IP (Intellectual Property) hırsızlığı ve sahtecilik tehditleri, donanım truva atı, derin öğrenme ağlarına yönelik güvenlik açıkları ve saldırılar olmak üzere 5 grupta incelenmiştir. Mimari ve sistem tehditleri; Güvenli Önyükleme Saldırıları, 'Firmware' Saldırıları, Dinamik Rastgele Erişimli Bellek Tehditleri, Önbellek Saldırıları, Spekülatif Çalıştırma Saldırıları ve Kodu Yeniden Kullanma Saldırıları (Code Reuse Attacks) başlıkları altında incelenmiştir. Gizli (Covert) ve Yan Kanallar; Zamanlama Kanalı, Güç Kanalı, Elektromanyetik ve Fotonik Kanallar, Arıza Enjeksiyonu olarak 4 başlıkta incelenmiştir. IP Hırsızlığı ve Sahtecilik Tehditleri başlığında çip tasarımlarının güvenlik tehditleri anlatılmıştır. Donanım Truva Atı kategorisinde Klasik Dijital Truva Atları, Umursamama Koşullarının (Don't Care Conditions) İstismarı, Analog Truva Atları ve saldırılar anlatılmıştır. Derin Öğrenme Ağlarına Yönelik

Güvenlik Açıkları ve Saldırılar; Çekişmeli (Adversarial) Örnekler, Donanım Yönelimli Saldırılar ve Model çıkartma saldırıları olarak farklı başlıklar altında incelenmiştir.

Yapılan çalışmalar incelendiğinde donanım güvenliği kapsamında donanıma yönelik saldırıların birçok farklı başlık altında toplandığı ve saldırı çeşitliliğin oldukça fazla olduğu görülmektedir. Ayrıca bu başlıklar da kendi içerisinde birçok alt kategoriye ayrılabilmektedir. Örneğin Gizli ve Yan Kanal Saldırıları altındaki Güç Analizi Saldırıları kendi içerisinde Basit Güç Analizi ve Farksal Güç Analizi olarak ayrılmakta ve benzer mantıkta işleyen ama farklı noktalardan hedefe saldıran yeni saldırı türleri de ortaya çıkmaktadır.

OWASP'ın (Open Web Application Security Project) yayınladığı IoT Saldırı Yüzey Alanları Projesi'nde saldırı yüzeyleri ve bu saldırı yüzeylerinin güvenlik açıkları tablo halinde sunulmuştur [32]. Projede yer alan tablo incelendiğinde donanım güvenliği ile alakalı olabilecek başlıkların listelendiği Tablo 2.1 oluşturulmuştur.

Tablo 2.1: Donanımı ilgilendirebilecek saldırı yüzeyleri ve güvenlik açıkları [32].

Saldırı Yüzeyi	Güvenlik Açığı

Cihaz Hafızası	• Hassas veri
	  ○ Okunabilir kullanıcı adları
	  ○ Okunabilir parolalar
	  ○ Şifreleme anahtarları
	  ○ 3. parti kimlik bilgileri

Cihaz Fiziksel Arayüzleri	• Firmware dışa çıkarma • Kullanıcı komut satırı arayüzü • Yönetici (Admin) komut satırı arayüzü • Yetki yükseltme • Güvensiz duruma resetleme • Depolanan medyanın silinmesi • Kurcalamaya karşı koruma • Hata ayıklama portları • Cihaz ID/Seri numarası teşhiri
Lokal Veri Sızıntısı	• Şifrelenmemiş veri • Keşfedilen anahtarlarla şifrelenmiş veriler • Veri bütünlüğü kontrollerinin eksikliği • Statik aynı şifreleme/şifre çözme anahtarının kullanımı
Donanım (Sensörler)	• Algılama ortamının manipülasyonu • Kurcalama (fiziksel olarak) • Zarar verme (fiziksel olarak)

Bu bölümde donanım güvenliğinde dikkat çeken ve sistem üzerinde büyük zarar veya bilgi kayıpları oluşturabilecek saldırı türlerine kısaca değinilmiştir. Bu saldırılar: Donanım Truva Atı, Fiziksel Arayüz Saldırıları ve Yan Kanal Saldırıları'dır.

2.2.1. Donanım Truva Atı

Bir IoT cihazının donanım tarafında güvenliğini sağlamak için öncelikle donanımı oluşturan gömülü sistemi güvenli hale getirmek gerekir. Gömülü sistemin güvenliği için ise burada kullanılan mikrodenetleyici, mikroişlemci, işlemci gibi entegre devrelerin güvenli olması gerekir. Entegre devreler Moore kanunun da etkisiyle her geçen gün daha çok transistör içermekte ve daha karmaşık hale gelmektedir. Tabi bu durum kullanıcılara hız ve işlem kapasitesinin artışı olarak yansısa da bir entegrenin üretimi birden fazla firma tarafından yapıldığından üretim aşamasında entegrenin tasarımının güvenliğinin sağlanması büyük önem taşımaktadır. Birden fazla firmanın üretimde bulunmasıyla entegre tasarımının dosyaları farklı firmalarda bulunmakta ve dışardan bir müdahaleye neden olabilmektedir. Bu müdahale ile entegre devrenin tasarımına zararlı bir devre eklenerek test aşamasında fark edilmediğinde entegrenin uzaktan kontrolü, bilgi sızdırma veya arızaya yol açarak sistemin çalışmasını engelleme gibi kötü durumlar ile karşılaşılabilmektedir. Bu tip donanım saldırılarından birisi donanım truva atlarıdır.

Donanım truva atları, bir çip tasarımının herhangi bir evresinde devreye entegre edilebilen, kötü amaçlı bir devre veya entegre devrenin kötü amaçlı işler için değiştirilmiş bir parçasıdır. Bu devre sistemin çıktısını değiştirebilir, çipe zarar verebilir veya çip içerisindeki bilgileri dışarıya sızdırabilir [26]. Donanım Truva atı devresi, milyarlarca transistör ve karmaşık bir tasarım içerisinde çok küçük bir devre olduğundan tespit edilmesi oldukça güçtür. Donanım truva atları çip çalışmaya başladıktan sonra aktif olabileceği gibi belirli durumlarda tetiklenerek de çalışmaya başlayabilir. İşlemcinin bu tipte bir donanım truva atı içermesi ilgili işlemcinin kullanıldığı sistemi tamamen güvensiz hale getirebilir.

2.2.2. Firmware Saldırıları

Bir IoT cihazı içerisinden ürün yazılımını yani 'firmware'ini çıkarmak, cihazın güvenliğini analiz ederken gerçekleştirilen çok önemli bir ilk adımdır. Firmware Saldırıları, cihazın donanım tarafındaki yazılımının güvenliğine odaklanmaktadır. Firmware donanım üzerinde çalışan ve entegrenin etkileşimini, davranışını kontrol eden bir yazılım olduğundan donanım güvenliği açısından oldukça önemlidir. Bu nedenle firmware çipin güvenliğini belirlemede

önemli bir rol oynamaktadır [31]. Gömülü sisteme ait yazılıma erişim cihazın nasıl çalıştığı ve içerisinde olabilecek güvenlik açıkları hakkında saldırgana bilgi sağlayabilmektedir. Ürüne veya yazılımına ait parolalar, gömülü anahtarlar gibi hassas veriler güvenliğin tam sağlanamadığı bir firmware içerisinde bulunabilir. Ayrıca firmware çıkarma için uygulanan saldırı vektörleri cihaza yazma erişimi sağlayarak firmware değişikliğini mümkün kılabilmektedir [33].

Cihaza ait firmware dosyası internet üzerinden ya da güncelleme esnasında ağı dinleyip ilgili paketi yakalayarak elde edilebileceği gibi donanım üzerinden de ürün yazılımına erişilebilir. Her cihaz farklı donanımlara ve üretim tekniklerine sahip olduğundan bu işlemin standart bir işlem haritası yoktur. Cihaz özelinde işlem yapmak gerekir. [34]'te yer alan projede piyasada bulunan birçok cihazın ürün yazılımına nasıl erişilebileceği proje sayfasında listelenen ürünler bazında adım adım anlatılmaktadır. Bu ürünler arasında akıllı ampullerden akıllı prizlere, akıllı bebek kameralarından buzdolabı ve televizyonlara kadar birçok akıllı cihaz bulunmaktadır. Bu cihazların ürün yazılımlarına nasıl erişilebildiği açık bir şekilde yazmakta ve kimi cihaz özelinde ise nasıl 'root' kullanıcısı olunacağı yer almaktadır.

Gömülü sistemin ana parçası olan mikrodenetleyiciler genellikle entegre hata testleri, bellek erişimleri ve entegre flaş çiplerini programlamak için hata ayıklama işlevleri içermektedir. Bu işlemler ve haberleşme işlemleri için mikrodenetleyiciler ve mikroişlemciler; UART (Universal Asynchronous Receiver Transmitter), JTAG (Joint Test Action Group), SWD (Serial Wire Debug), SPI (Serial Peripheral Interface) ve I2C (Inter-Integrated Circuit) gibi arabirimler kullanmaktadır. Donanım içerisindeki yazılıma erişmek, bellek içeriğini okumak veya donanım içerisindeki işletim sisteminin komut satırına erişim sağlamak için birçok yöntem olduğu gibi bu hata ayıklama ve seri haberleşme arayüzlerinden de yararlanılmaktadır. Firmware çıkarmanın mümkün olduğu çoğu durumda aynı yöntemle firmware modifikasyonu da gerçekleştirilebilir [33]. Bu durum firmware çıkarımından daha kötü sonuçlar doğurabilir. Kötü amaçlı firmware modifikasyonu yapılmış bir cihaz tekrar başka bir kullanıcıya satılabilir ve saldırgan cihaz içerisine yerleştirdiği uzaktan kontrol veya dinleme yazılımı ile satın alan kullanıcının hassas verilerine erişim sağlayabilir.

Donanım kullanılarak firmware elde etmek için UART arayüzü kullanılmak istendiğinde saldırgan donanım üzerinde UART pinlerini tespit edip bir UART-bilgisayar haberleşmesi sağlayan cihaz ile UART üzerinden cihaza erişim sağlayabilir. Bu saldırıda eğer UART pinleri

açıksa saldırgan terminale erişebilir ya da cihazın o an yaptığı işlemler hakkındaki kayıtları inceleyebilir. UART erişiminde komut satırında parola koruması olmaması veya basit bir parola politikasının olması saldırganın ürün içerisindeki yazılıma erişmesiyle sonuçlanabilmektedir. Bu işlem JTAG veya diğer arayüzler için farklı adımlar ve araçlar gerektirse de önlem alınmadığında ürün yazılımı veya hassas bilgiler saldırganların eline geçebilmektedir.

[33]'te UART ve JTAG üzerinden nasıl firmware erişimi yapılabileceği, hafızanın nasıl okunabileceği anlatılmış ve piyasada bulunan birçok cihaz üzerinde yapılan firmware elde etme çalışmaları anlatılmıştır. Ayrıca bu saldırılara karşı alınabilecek önlemler de aktarılmıştır.

2.2.3. Yan Kanal Saldırıları

IoT donanımları kriptografik işlemleri gerçekleştirirken dışarıya güç tüketimi, elektromanyetik yayılım veya işlem gecikmesi gibi istemsiz bilgi sızdırabilmektedir. Yan kanal saldırıları saldırganların bu sızıntıları ölçerek ve analiz ederek gerçekleştirdikleri ve sonucunda donanım üzerinden özel anahtarları elde edebildikleri ya da hassas verilere ulaşabildikleri invaziv olmayan bir saldırı türüdür. Yan kanal saldırıları cihazın işleyişine etki etme durumlarına göre ikiye ayrılmaktadır. Bunlar aktif ve pasif saldırılardır. Aktif saldırılar, işlemciyi hata vermeye zorlayarak bilgi edinme gibi cihaz işleyişinin değiştirildiği saldırılardır. Pasif saldırılar ise saldırganın cihaz davranışını bozmadan basitçe gözlem yaptığı ve yan kanal çıktılarından faydalandığı saldırılardır. Bu tip saldırılarda saldırganlar yan kanal bilgilerini elde etmek ve analiz etmek için osiloskop, sinyal analiz yazılımı, lojik analizör gibi çeşitli ekipmanlara ihtiyaç duyarlar [35].

Birçok kaynakta [28, 29, 31] Yan Kanal Saldırıları içerisine aşağıdaki saldırı çeşitleri dahil edilmektedir:

- Güç Analizi Saldırıları
- Elektromanyetik Analiz Saldırıları
- Zamanlama Analizi Saldırıları
- Akustik Analiz Saldırıları
- Hata Saldırıları

Yan kanal saldırıları kriptografik işlem gerçekleştiren veya parola koruması içeren cihazların güvenliği ciddi tehlike oluşturmaktadır. Hem donanım tarafında hem de yazılım tarafında gerekli önlemlerin alınmaması durumunda şifreleme anahtarı gibi sistem için kritik veriler saldırganların eline geçebilmektedir. Bölüm 3.1'de tez çalışması kapsamında yapılacak yan kanal saldırıları detaylandırılarak anlatılmıştır.

2.2.4. Donanım Güvenliği Saldırı Örnekleri

IoT ile birlikte cihazların internete bağlanmasının güvenlik konusunda bağlanan cihazın donanımını da ilgilendirdiği ve gelebilecek saldırı çeşitleri önceki bölümlerde anlatılmıştır.

[36]'da yapılan çalışmada otomotiv gömülü sistemlerinin belleğinin çalışması, olası tehditler ve güvenlik açıkları değerlendirilmiştir. Otomobilin elektronik kontrol ünitelerindeki düşük seviyeli bellek bölümlerinin hedef alındığı çalışmada ayrıca belleklere yönelik genel saldırı stratejilerinden de bahsedilmiştir. Gömülü sistem üzerinde ilk saldırı olarak Yığın Tabanlı Arabellek Taşması denenmiş ve saldırı sırasında karşılaşılan zorluklar ve sınırlamalar açıklanmıştır. Çalışmada bahsedilen istismarlar dikkatli kullanıldığında saldırganın 'root' seviyesinde yetkiler elde edebileceğinden bahsedilmiştir. Gömülü sistem üzerinde fiziksel bellek adres aralığına erişmek, hazırlanan kodu RAM'de (Random Access Memory – Rastgele Erişimli Bellek) yürütmek ve kısıtlı bellek alanına erişmek için Hata Ayıklayıcıları Kullanarak Fiziksel Belleğe Erişim saldırısı denenmiştir. Bu saldırıda donanım üzerindeki UART portu kullanılarak donanıma erişim sağlanmış ve verilerin başka bir cihaza iletilip alınabildiği bilgisi paylaşılmıştır. [37]'de yer alan haberde siber güvenlik araştırmacılarının STMicroelectronics veya firmware tabanlı Intel TPM'ler (Trusted Platform Module) tarafından üretilen TPM çipleri içerisinde korunan kriptografik anahtarların elde edilmesine olanak sağlayabilecek iki CPU güvenlik açığı açıkladıkları paylaşılmıştır. Zamanlama tabanlı yan kanal saldırısından yararlanılan bu güvenlik açığı ile zamanlama verileri toplanarak özel anahtarlar üzerinde test edilmiş ve anahtar elde edilmiştir. Aynı zamanda araştırmacıların Infineon ve Nuvoton tarafından üretilen TPM çözümlerinin de zamanlama sızıntısı konusunda savunmasız olduğunu keşfettikleri bilgisi haberde yer almaktadır. [38]'de yer alan haberde Apple'ın M1 işlemci çipsetlerine yönelik PACMAN isimli yeni bir donanım saldırısı ve etkileri paylaşılmıştır. Geçmiş yıllarda, günümüzde kullanılan işlemcilerin çoğunu hedef alan Meltdown ve Spectre açıkları yayınlanmıştır. Bu açıklar kullanılarak cihazda çalışan bir kod, erişim izni olmayan hafıza alanlarına erişebilmektedir. Bu durum gizli anahtarlar ve parolalar olmak üzere hassas verilerin tehlikeye girmesine sebep olabilmektedir [39].

2.3. KRİPTOGRAFİ

Artan siber saldırılar neticesinde haberleşme altyapısının güvenliği ön plana çıkmaktadır. Gizli bir konuşma yapmak isteyen 2 taraf arasına kötü niyetli bir 3. taraf girdiğinde aralarındaki haberleşmeyi dinleyerek konuşulanlar hakkında bilgi sahibi olabilmektedir. Bu gibi durumlarda haberleşme dinlense dahi araya giren kötü tarafın içerikleri elde etse bile anlamamasını sağlamak gerekir. Bu durum kriptografi ile birlikte verilerin şifrelenerek karşı tarafa iletilmesiyle güvenli hale gelmiş ve araya giren kötü niyetli kişilerin anlamlı verilere erişmesinin önüne geçilmiştir.

Herhangi bir haberleşme ortamında, aşağıdakiler de dahil olmak üzere bazı özel güvenlik gereksinimlerinin karşılanması gerekmektedir. Kriptografinin 5 temel işlevi bulunmaktadır [40]:

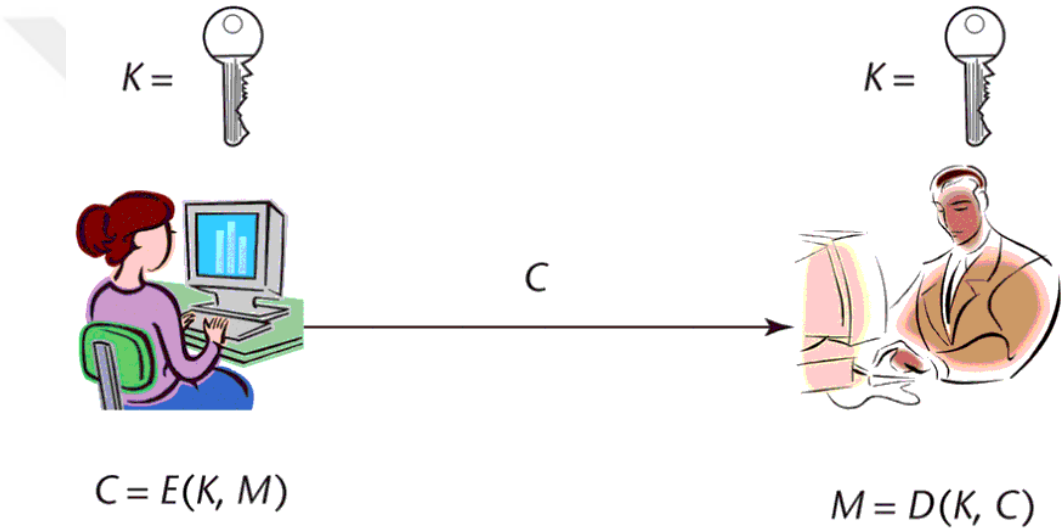
- Kimlik Doğrulama (Authentication): Kişinin kimliğini kanıtlama süreci
- Mahremiyet/Gizlilik (Privacy/Confidentiality): Hedeflenen alıcı dışında hiç kimsenin mesajı okuyamamasını sağlamak
- Bütünlük (Integrity): Alıcıya, alınan orjinal mesajın hiçbir şekilde değiştirilmediğini garanti etmek
- Reddetmeme (Non-repudiation): Gönderenin gerçekten bu mesajı gönderdiğini kanıtlayan bir mekanizma
- Anahtar değişimi: Şifre anahtarlarının gönderici ve alıcı arasında paylaşıldığı yöntem

Bu maddeler incelendiğinde kriptografinin sadece verilerin okunmaması veya değiştirilmemesi konusunda değil, kullanıcının kimlik doğrulaması gibi konularda da kullanılmaktadır [40].

Şifreleme işlemi, bir açık metnin (plaintext) bir veya birden fazla anahtar (key) kullanılarak şifreli bir metin (ciphertext) haline getirilmesi ve şifreli metnin aynı veya farklı anahtar kullanılarak tekrar açık metin haline döndürülmesi işlemidir. Anahtar, sayısal veya alfa numerik bir metin veya özel bir sembol olabilir. Anahtar, şifrelemenin düz metin üzerinde gerçekleştiği anda ve şifrenin çözüldüğü sırada şifreli metin üzerinde kullanılır. Şifreleme algoritmasının güvenliği doğrudan anahtara bağlı olduğu için kriptografide anahtar seçimi çok önemlidir [41]. Şifrelemede kullanılan anahtarların sayısı ve kullanım şekline göre (şifreleme/şifre çözme)

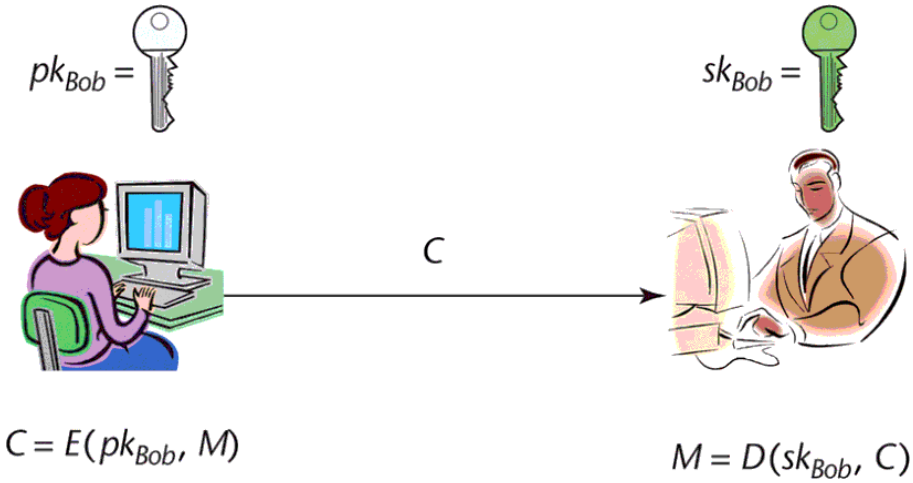
simetrik şifreleme ve asimetrik şifreleme olarak farklı kategorilere ayrılan şifreleme algoritmaları bulunmaktadır [40].

Simetrik Şifreleme; şifreleme işlemi esnasında kullanılan anahtar ile şifreyi çözme esnasında kullanılan anahtarın aynı olduğu şifreleme metodudur. Gönderici taraf anahtarı kullanarak metni şifreler, alıcı taraf aynı anahtarı kullanarak şifrelenmiş metni orijinal haline getirir. Simetrik Şifreleme; Blok ve Akış Şifreleme olmak üzere kendi içerisinde ikiye ayrılmaktadır. AES (Advanced Encryption Standard - Gelişmiş Şifreleme Standardı) ve DES (Data Encryption Standard - Veri Şifreleme Standardı) blok şifrelemeye, RC4 (Rivest Cipher 4) akış şifrelemeye örnek olarak verilebilir [41].



Şekil 2.10: Simetrik Şifreleme. Alice M mesajını K anahtarını kullanarak E algoritmasıyla şifreleyip C şifrelenmiş metnini Bob'a gönderiyor. Bob aynı K anahtarını kullanarak C şifreli metninin şifresini çözerek M mesajını elde ediyor [42].

Asimetrik Şifreleme ise iki farklı anahtar içerir. Bu anahtarlardan birisi herkes tarafından bilenen 'public key' (açık anahtar), diğeri ise sadece ilgili kişinin şifrelenmiş metni açabileceği (yani 'public key' ile şifrelenmiş metni açabileceği) 'private key' (gizli anahtar)'dir. Burada gönderici, metni alıcının 'public key''i ile şifreleyerek iletir. Alıcı 'private key''ini kullanarak şifrelenmiş metni açar. RSA (Rivest-Shamir-Adleman) asimetrik şifrelemeye örnek verilebilir. Asimetrik şifreleme teknikleri, simetrik şifrelemeden çok daha yavaştır, bu nedenle büyük miktarda veriyi şifrelemeye çalışırken pratik kalmamaktadır [41].



Şekil 2.11: Açık Anahtarlı Şifreleme. Asimetrik Şifreleme. Alice M mesajını Bob'a ait 'public' anahtarını (pk) kullanarak E şifreleme algoritmasıyla şifreleyip C şifreli metnini Bob'a gönderiyor. Bob, C şifreli metninin kendine ait olan 'private' anahtar (sk) ile şifresini çözüp M mesajını elde ediyor [42].

DES, 1970'li yıllarda IBM tarafından 64 bitlik bir anahtar kullanarak 64 bitlik veri bloklarını şifrelemek ve şifresini çözmek için tasarlanmış simetrik anahtarlı blok şifreleme algoritmasıdır. Şifreleme algoritması şifrelenecek metin bloklarını anahtardan elde edilen değerlerle iç içe geçiren 16 yinelemeden oluşur. 64 bit girişi bir dizi işlemden geçirerek 64 bit çıkışa dönüştürür. Şifre çözmek için de aynı anahtarla aynı adımlar uygulanır. Zaman içerisinde DES'i güvensiz hale getirecek birçok saldırı ve yöntem ortaya çıkmıştır. Yine de birçok uygulamada hala kullanılmaktadır [41].

AES, 2001 yılında NIST (National Institute of Standards and Technology - Ulusal Standartlar ve Teknoloji Enstitüsü) tarafından DES'in yerini alması için önerilen simetrik anahtarlı blok şifreleme algoritmasıdır. 128, 192 ve 256 anahtar uzunluğunu destekleyen AES algoritması anahtar uzunluğuna göre AES-128, AES-192 ve AES-256 olarak adlandırılır. AES algoritması şifreleme ve şifre çözme işlemi esnasında 128 bit anahtar için 10, 192 bit anahtar için 12 ve 256 bit anahtar için 14 döngüden geçmektedir. Algoritma 4 temel operasyonel bloğa bölünebilen 128 bitlik bir veri uzunluğuna izin verir. Bu bloklar bayt dizisi olarak ele alınır. AES hem şifreleme hem de şifre çözme işlemi için 'tur anahtarı ekleme' aşamasıyla başlar ve şifreleme turlarından geçer. Bu turların her birinde dört işlem gerçekleştirilir: Bayt Değiştirme, Satır Kaydırma, Sütun Karıştırma ve Adım Anahtarı Ekleme. Son turda sütun karıştırma dönüşümü yoktur. Şifre çözme işlemi ise şifrelemenin tersine işlemidir ve tersine fonksiyonlar kullanılır: Ters Bayt Değiştirme, Ters Satır Kaydırma ve Ters Sütun Karıştırma gibi. Ayrıca yapılan

incelemelerde AES algoritmasının hız, zaman ve verim konusunda fazlasıyla verimli olduğu görülmüştür [41].

RSA, 1978 yılında Ron Rivest, Adi Shamir ve Leonard Adleman tarafından oluşturulan ve soyadlarının baş harflerini taşıyan asimetrik (açık anahtarlı) bir şifreleme algoritmasıdır. Anahtar değişimi, dijital imzalar ve verilerin şifrelenmesi gibi işlemlerde yaygın olarak bilinen bir şifreleme algoritmasıdır. Algoritma şifreleme ve şifre çözme işlemleri için AES ve DES'in aksine 2 farklı anahtar kullanmaktadır. Mesaj alıcısının 2 anahtarı bulunmaktadır. Bunun bir tanesi herkesle paylaştığı 'public' (açık) anahtarı diğeri ise sadece kendisinde olan 'private' (gizli) anahtarıdır. Gönderici alıcıya ait public anahtar ile ileteceği mesajı şifreleyerek alıcıya gönderir. Alıcı sadece kendisinde bulunan private anahtarı kullanarak göndericinin public anahtar ile şifrelediği metnin şifresini çözerek mesaja ulaşır. Şifreleme işlemi 3 adımda oluşturulur: anahtar üretme, şifreleme ve şifreyi çözme. Anahtar üretmek için çok büyük rastgele iki asal sayı p ve q belirlenir ve bu değerler kullanılarak public ve private anahtarlar üretilir. Burada p ve q değerleri küçük sayılar seçildiğinde şifreleme güvenliği çok zayıflar ve şifrelemeye yönelik saldırılar sonucunda şifre çözülebilir. Eğer p ve q çok büyük seçilirse işlem süresi uzar ve diğer şifreleme algoritmalarına göre performansı düşer [41].

Şifreleme algoritmaları bilgi güvenliğinin önemli olduğu her yerde kullanılmaktadır. Mesajların güvenliğini ve bütünlüğünü sağlamak için mesaj ve mail uygulamalarında, haberleşmenin güvenliğini sağlamak için ağ haberleşmelerinde, akıllı kartlarda, ödeme sistemlerinde ve güvenlik gerektiren daha birçok yerde ihtiyaç duyulan güvenliği sağlayacak ve performansı kötü etkilemeyecek ilgili şifreleme algoritmaları kullanılmaktadır. IoT cihazlarda internete bağlanan cihazın donanımında verilerin uzak sunucuya güvenli ve bütün halde iletilmesi için donanım özelinde şifreleme yapılmakta ve verilerin iletimi şifrelenerek sağlanmaktadır. Ayrıca donanım içerisindeki verilerin, hafızanın veya yazılımların da şifrelemesi yapılarak dışarıdan müdahalelere karşı koruma sağlanmaktadır.

Şifreleme algoritmaları dışında kriptografide kullanılan veri bütünlüğü sağlayan uygulamalardan birisi de hash fonksiyonlarıdır. Hash fonksiyonları mesaj özeti olarak da bilinmektedir. Anahtar kullanılmayan ve tek yönlü işlem gerçekleştirilen hash işlemlerinde bir verinin alınan hash değeri verinin boyutu, içeriği fark etmeksizin sabit uzunlukta olup tekrar eski haline döndürülemez. Hash fonksiyonları verinin en ufak bölümü değişse dahi elde edilen

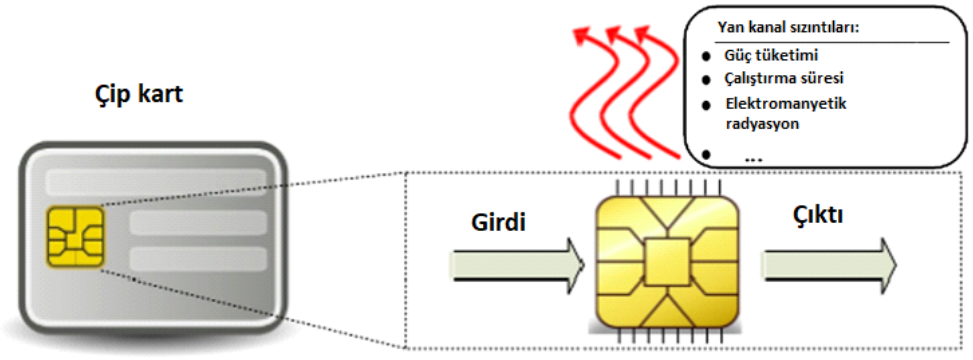
ilk sonuçtan çok farklı sonuç üretmektedir. Bu sebeple genellikle bir dosyanın saldırganlar tarafından değiştirilmediğinden emin olmak için kullanılmaktadır [40].

Saldırganlar cihazlarda alınan şifreleme önlemine yönelik saldırı metotları geliştirmiş ve şifreleme adımını geçmeyi ya da şifreleme anahtarını elde etmeye çalışmışlardır. Literatürde de şifreleme güvenliği konusunda birçok çalışma paylaşılmış ve araştırmacılar hem şifreleme algoritmalarının hem de şifrelemenin uygulanmadığı yazılım veya donanımların güvenlik durumlarını incelemişlerdir. Şifreleme güvenliği konusunda birçok çalışmada şifreleme algoritmasının kendisine yönelik kriptanaliz çalışmaları yapılmış ve şifreleme algoritmalarının güvenilirlikleri test edilmiştir. [43]'te yer alan yazıda diferansiyel kriptanaliz uygulanarak 6 turluk DES şifrelemesi kırılmaya çalışılmış ve yapılan işlemin sonunda anahtar elde edilmiştir. Şifreleme güvenliği kapsamında yapılan bir diğer saldırılar ise şifrelemenin gerçekleştiği donanımlara yönelik saldırılardır. Bu saldırılar donanımın harcadığı güç miktarı veya elektromanyetik yayılımı gibi yan kanal bilgilerini elde edip analiz ederek şifreleme anahtarını elde etmek için yapılmaktadır. [44]'te kriptografik işlem gerçekleşen işlemcinin güç tüketimi analiz edilip gizli bilgilerin elde edilebildiği bir çalışma anlatılmıştır. Bu tip saldırılar literatürde yan kanal saldırıları olarak geçmektedir ve her geçen gün elde edilen yan kanal bilgilerine göre yeni saldırı yöntemleri geliştirilmektedir.

3. YÖNTEM

3.1. YAN KANAL SALDIRILARI

Güvenlik araştırmacıları ve saldırganlar bir cihazdaki şifrelenmiş verileri elde etmek ya da şifrelenmiş haberleşme yapısını ele geçirmek için şifreleme algoritmasındaki istatistiksel veya matematiksel zayıflıkları analiz etmek yerine şifreleme algoritmasının çalıştığı donanımı hedeflerine almışlar ve donanım üzerinden şifreleme anahtarını çekmeye çalışmışlardır. Bir donanım üzerinde şifreleme işlemi gerçekleşirken bu işlem için bir şifreleme anahtarı ve şifrelenecek bir veri gerekir. Şifreleme işlemi gerçekleşirken işlemci yaptığı işlemlerin sonucunu üretmenin dışında dışarıya istenmeyen çıkışlar da verebilmektedir. Yan kanal saldırıları şifrelemenin gerçekleştiği donanımın hedeflendiği invaziv olmayan saldırı türüdür. Bu saldırılar hedef donanımın, üzerindeki çeşitli kaynaklardan veya kanallardan dışarıya sızdırdığı güç tüketimi bilgisi, elektromanyetik yayılım, hesaplama için geçen süre bilgisi veya sistemin çalışmasına bağlı çıkardığı sesler gibi fiziksel bilgilerden yararlanmaktadır. Bu kanallara "yan kanallar" denmektedir. Yan kanal sızıntılarındaki bilgi, şifreleme işleminin yürütülmesi sırasında hesaplanan ara değerlere bağlı olmasının yanında hesaplamanın girişleri ve şifreleme anahtarı ile ilişkilidir [29, 45]. Şekil 3.1'de bir cihazın yan kanal bilgilerini nasıl sızdırdığı gösterilmektedir.



Şekil 3.1: Kriptografik donanımın çalışması esnasında ortaya çıkan yan kanal sızıntıları [29].

Saldırgan, donanımın içerisinden bu yan kanal parametrelerini, temin edebileceği ekipmanlarla birkaç dakika ile birkaç saat arasında değişen çok kısa bir süre içinde gözlemleyerek ve analiz

ederek gizli anahtarı çıkarabilir. Bu nedenle yan kanal saldırıları, saldırganların yan kanalları elde edebileceği kriptografik cihazlar için özellikle akıllı kartlar ve IoT cihazları için büyük tehdit oluşturmaktadır. Yan kanal saldırılarının bildirilen ilk örneği 1965 yılında bir devlet kurumuna yapılan ve her gün sıfırlanan bir anahtar kullanarak şifreli metinler üreten bir kriptografik makineye uygulanan saldırıdır. Makine tarafından üretilen tıklama seslerinin sayısını anahtarla ilişkilendiren saldırganlar modülün sesini kaydederek gizli anahtarı elde etmişlerdir. Bu örnekten günümüze kadar geçen sürede yan kanal saldırıları oldukça genişlemiş ve birçok yeni yan kanal analizi yöntemi geliştirilmiştir [29].

Saldırı çeşitleri yapılan saldırının cihazın çalışmasını etkileme ve saldırganın cihaz üzerinde sahip olabileceği kontrol seviyesine bağlı olarak pasif ve aktif saldırılar olarak iki ana başlıkta sınıflandırılabilir. Pasif saldırılar (güç analizi, elektromanyetik analiz, zamanlama analizi gibi) cihazın işlevselliğine ve çalışmasına müdahale edilmeden gerçekleştirilen saldırılardır. Saldırı, sistem normal şekilde çalışırken başlatılır ve cihazın yan kanal bilgileri (güç tüketimi, işlemlerin zamanlama gecikmesi gibi) yapılan gözlemlerle elde edilerek şifreleme anahtarı elde edilmeye çalışılır. Aktif saldırılarda (hata oluşturma saldırıları gibi) ise cihaza müdahale yapılarak gerçekleştirilir. Saldırgan harici bir arayüz kullanarak ya da çevresel koşullar aracılığıyla cihaza müdahalede bulunur ve hedef cihazın davranışını etkiler. Bu etkiyle cihazı hataya zorlayarak veya yaptığı müdahalelerle cihazın davranışını kontrol etmeye çalışarak elde ettiği yan kanal bilgileriyle güvenlik mekanizmasını atlatmaya veya şifreleme anahtarını elde etmeye çalışır [29, 45].

Yan kanal saldırılarında kullanılan yan kanal bilgisine göre farklı saldırı yöntemleri mevcuttur. Literatürde birçok kaynakta [28, 29, 31] aşağıda listelenen saldırı çeşitleri yan kanal saldırılarına dahil edilmektedir:

- Güç Analizi Saldırıları
- Elektromanyetik Analiz Saldırıları
- Zamanlama Analizi Saldırıları
- Akustik Analiz Saldırıları
- Hata Saldırıları

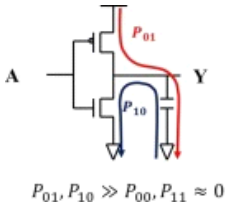
Yan kanal saldırıları birçok farklı şekilde gerçekleştirilebilmektedir. Basit gözleme dayalı analizlerin yapıldığı saldırıların yanında istatistiksel yöntemler kullanılarak yapılan saldırılar da şifreleme anahtarını elde etmek için kullanılmaktadır. Bu metotlar saldırılar kendi içlerinde basit gözlem ve istatistiksel yöntem şeklinde sınıflandırmaları yapılabilir. Aynı şekilde hata saldırıları için yan kanal sinyalini üretmek için kullanılan yönteme göre gerilim ve 'clock' (saat) şeklinde sınıflandırma yapılabilir. Yaygın kullanılan yan kanal saldırılarının (güç analizi, elektromanyetik analiz, zamanlama analizi) yanında donanımın yaptığı işlemler hakkında bilgi sızdırabilen, yayılan ses, sıcaklık ve titreşim gibi yan kanal sinyalleri de bulunmaktadır [29]. Yayılan ses bilgisi kullanılarak yapılan akustik analiz saldırılarına, yan kanal saldırıları konusunda 1965'te bildirilen ilk saldırı olan, makine tarafından üretilen tıklama seslerinin sayısının anahtarla ilişkilendirilmesiyle yapılan saldırı örnek olarak verilebilir. Ayrıca [46]'da akustik analiz saldırısıyla hassas mikrofonlar kullanarak 4096 bit RSA anahtarı kullanan GnuPG yazılımından anahtarın elde edildiği çalışma anlatılmıştır. Güvenli olması beklenen bir cihazın çalışacağı saha bilgisi, içerdiği bilgiler ve bilgilerin kritiklik durumlarına göre güvenlik seviyesi belirlenmeli, düşman ya da saldırgan eline geçmemesi gereken kritik bilgilerin korunması için donanımdaki bu yan kanal parametrelerinin değerlendirilmesi ve önlemlerin alınması gerekmektedir.

3.1.1. Güç Analizi Saldırıları

Elektronik cihazlar çalıştıkları esnada belirli bir elektriksel güce ihtiyaç duyarlar. Şifreleme işlemi gerçekleştiren bir işlemcinin bu işlemi gerçekleştirirken bir güç tüketimi olmaktadır. İşlemcinin güç tüketimi işlediği verilere ve yaptığı işlemlere bağlıdır. Güç analizi saldırılarının temel fikri kriptografik işlem gerçekleştiren donanımın güç tüketimini analiz ederek yaptığı işlemler hakkında bilgi toplamak ve gizli anahtara ulaşmaya çalışmaktadır. Bu tip bir saldırı esnasında cihazın harcadığı güç ölçümleneceği için cihaza fiziksel erişim sağlanması gerekmektedir [29, 45].

Kullanılan entegre devreler, gerilim kontrollü anahtarlama yapan transistörlerden oluşur. Günümüzde sayısal entegre devreleri oluşturmak için en yaygın kullanılan teknolojilerden birisi CMOS (Complementary Metal Oxide Semiconductor) teknolojisidir. Bu nedenle bir işlemcinin şifreleme esnasında güç tüketimini analiz etmek için CMOS devrelerinin güç tüketimini incelemek gerekir. CMOS devrelerinin güç tüketimi statik ve dinamik olmak üzere iki bileşenden oluşmaktadır. Statik güç tüketimi transistörlerin kaçak akımından kaynakları ve bu nedenle devre tasarımına bağlıdır. Dinamik güç tüketimi ise transistörlerin anahtarlanmasından

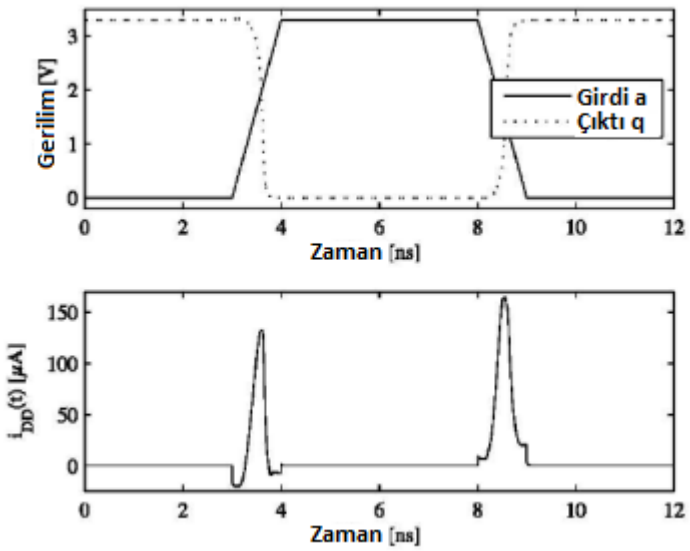
kaynaklanır ve bu nedenle devrede işlenen verilere ve devrenin yaptığı işlemlere bağlıdır. Statik güç bir işlem süresince çoğunlukla sabit olduğundan devrenin toplam güç tüketimindeki değişiklik dinamik güç tüketiminden kaynaklanmaktadır. Bir CMOS hücresi 0'dan 1'e veya 1'den 0'a durum değişimi yaşadığında anahtarlama meydana gelir ve güç tüketilir [47]. Şekil 3.2'de bir CMOS eviricinin durum değişimleri gösterilmiştir.



A	Y	Power
$0 \rightarrow 0$	$1 \rightarrow 1$	P_{11}
$\mathbf{0 \rightarrow 1}$	$\mathbf{1 \rightarrow 0}$	$\boldsymbol{P_{10}}$
$\mathbf{1 \rightarrow 0}$	$\mathbf{0 \rightarrow 1}$	$\boldsymbol{P_{01}}$
$1 \rightarrow 1$	$0 \rightarrow 0$	P_{00}

Şekil 3.2: Bir CMOS eviricinin durum değişimleri [29].

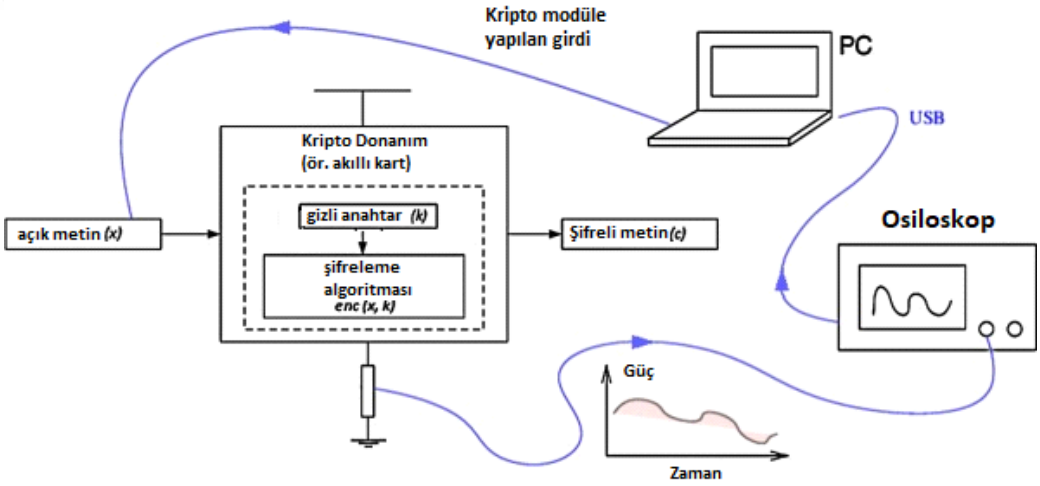
CMOS eviricideki anahtarlama işlemlerinde durum geçişlerinin harcadıkları güçler arasında değişiklikler olmaktadır. Şekil 3.2'de yer alan devrede 0'dan 1'e ve 1'den 0'a olan değişimlerin yolları gösterilmiştir. Çıkışa bağlı kondansatör çıkış değeri anahtarlandığında şarj veya deşarj olduğundan güç tüketimi diğer durumlardan daha büyük olacaktır. 0'dan 0'a veya 1'den 1'e olan durum değişimlerinde şarj veya deşarj aktivitesi olmadığından güç tüketimi neredeyse sıfır olmaktadır [29]. Şekil 3.3'te bir CMOS eviricinin giriş çıkış sinyalleri ve anlık akım değerleri gösterilmiştir.



Şekil 3.3: Bir CMOS eviricinin anahtarlama esnasında dinamik güç tüketimi [47].

Şekil 3.3'te görüldüğü gibi giriş sinyalinde 0'dan 1'e ve 1'den 0'a geçişlerin olduğu anahtarlama gerçekleşen anlarda devrenin çektiği akım anlık olarak yükselmektedir. Bu bilgiler kullanılarak işlemcinin güç tüketiminden yapılan işlemin çıktısı veya girdisi hakkında bilgi edinilebilir. Güç analizi saldırıları bu güç tüketimi yan kanal bilgisinden yararlanır ve saldırgana kriptografik işlem esnasında kullandığı gizli anahtar hakkında bilgi verebilir.

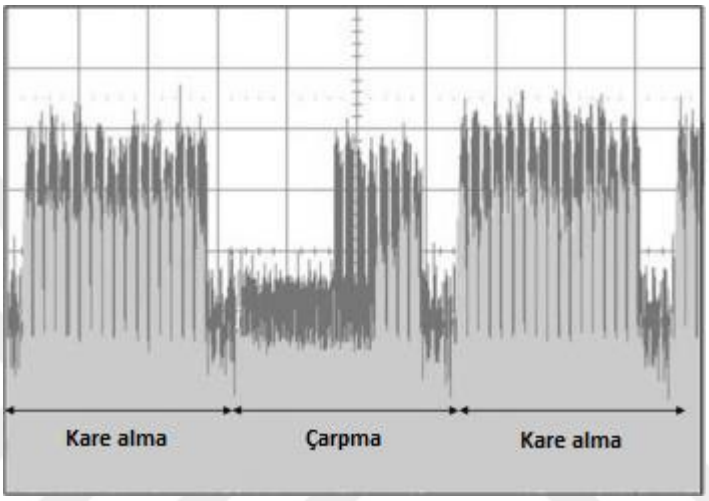
Güç saldırısı ile şifreleme anahtarını elde etmek için öncelikle işlemcinin güç sinyallerini yakalamak gerekir. Kriptografik işlem gerçekleştiren işlemcinin güç veya toprak girişine seri olarak yerleştiren direnç üzerinden osiloskop yardımıyla güç tüketim izleri elde edilebilir. Osiloskoptan bilgisayara aktarılan sinyal verileri analiz edilerek işlem gerçekleştirilir. Şekil 3.4'te bir güç analizi saldırısının süreci gösterilmektedir.



Şekil 3.4: Kriptografik işlem gerçekleştiren bir donanıma yapılan güç analizi saldırısının kurulumu [29].

Güç izi toplama işlemi donanım çalışırken gerçekleştiğinden gürültü içerebilmektedir. Bu gürültü diğer modüllerin anahtarlama aktivitesinden kaynaklandığı gibi çevresel etkilerden kaynaklanan doğal elektrik gürültüsü de olabilmektedir. İşlem yapılmak istenen güç izinin sayısı donanıma, kullanılan algoritmaya ve saldırı tipine bağlı olarak tek bir güç izinden milyonlarca güç izine kadar değişebilir. Bir sistemin güç tüketimi doğrudan gözlemlenip anahtara ilişkin bilgiler elde edilebilirken istatistiksel analiz kullanarak da şifreleme anahtarına ulaşılabilir. Bu işlemler bakımından güç analizi saldırıları farklı kategorilere ayrılmaktadır. Bunlar: Basit Güç Analizi (Simple Power Analysis - SPA), Diferansiyel Güç Analizi (Differential Power Analysis - DPA) ve Korelasyon Güç Analizi (Correlation Power Analysis - CPA) [29].

Basit güç analizi kriptografik işlemler sırasında elde edilen güç tüketimi ölçümlerini doğrudan yorumlamayı içeren bir tekniktir. SPA, anahtara bağlı işlemlerin gerçekleştiği bir güç izinden anahtar bitlerini elde etmek için kullanılabilir [44]. Bir işlemcideki her işlem güç izinde görsel olarak fark edilebilen bir desene neden olur. Bu desenler yapılan işlemler hakkında saldırgana bilgi verebilmektedir. Şekil 3.5'te yer alan görselde RSA algoritması çalıştıran işlemcinin güç izi gösterilmektedir. RSA algoritmasında yer alan kare alma ve çarpma işlemlerine karşılık gelen güç izleri sinyal üzerinden görülebilmektedir.



Şekil 3.5: RSA şifre çözme işleminde modüler üs alma güç izlerinin bir bölümü [29].

Diferansiyel güç analizi, verilere bağlı korelasyonları belirlemek için ölçüm setlerini analiz etmeye yönelik istatistiksel bir yöntemdir [48]. DPA saldırıları, gizli anahtara ilişkin bilgileri çıkarmak için istatistiksel analiz kullanır. Saldırgan, birçok giriş için kriptografik cihazın güç tüketimini izler ve daha sonra toplanan güç sinyali verilerini istatistiksel olarak analiz eder. Saldırgan, istatistiksel analiz sonuçlarını kullanarak gizli anahtarı çıkarabilmektedir [44]. Bu tip saldırıların gerçekleştirilmesi için donanım hakkında önceden bilgi sahibi olmak gerekmez ve bu saldırı yönteminde istatistiksel analiz yapıldığından gürültülü bir ortamda kaliteli sinyaller elde edilebilir. Bu işlemi gerçekleştirmek için basit güç analizine göre çok fazla sayıda güç izine ihtiyaç duyulmaktadır. Veri sayısının artışı saldırıyı daha güçlü hale getirmektedir [29].

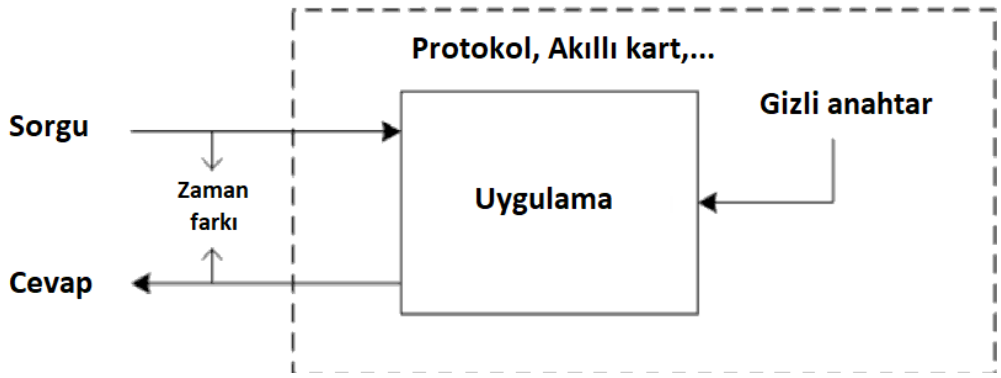
Korelasyon güç analizi, güç tüketimi ile hedef fonksiyonun (örneğin AES şifrelemedeki S kutusu işleminin çıktısı) Hamming mesafesi veya Hamming ağırlığı arasındaki korelasyondan yararlanan gelişmiş bir güç analizi saldırısıdır. İstatistiksel bir saldırı türüdür ve verileri

ilişkilendirmek için Pearson korelasyon katsayısını kullanır. DPA ile karşılaştırıldığında daha az sayıda güç izi gerektirmesi gibi avantajları bulunmaktadır [29, 47].

Literatürde bu konuda birçok çalışma yapılmış ve farklı saldırı yöntemleri denenmiştir. Bu saldırıların ilki olan [44]'te güç analizi gündeme gelmiş ve basit güç analizi ve diferansiyel güç analizi açıklanarak cihazlardan gizli anahtarları elde etme yöntemleri anlatılmıştır. [47]'de içerisinde AES şifreleme işlemi gerçekleştiren PIC mikrodenetleyici üzerinde korelasyon güç analizi çalışması yapılmıştır. [48]'de anahtarları çıkarmak için güç tüketimi ve diğer yan kanallardan sızan bilgilerin nasıl analiz edilebileceği anlatılmıştır. [49]'da RSA şifreleme gerçekleştiren donanımı diferansiyel güç analizi saldırılarına karşı dayanaklı kılan yöntem anlatılmıştır.

3.1.2. Zamanlama Analizi

Donanım içerisinde gerçekleşen işlemler belirli bir sürede hesaplanır ve tamamlanırlar. Yapılan işlemlerin süresi sistemin giriş değerlerine göre değişiklik gösterebilmektedir. İşlemlerin sisteme girilen girişler sonrasındaki yürütme sürelerindeki farklılıklar yapılan işlemler ve girişler konusunda bilgi verebilmektedir. Zamanlama analizi saldırılarında saldırganlar bu süre bilgisini analiz ederek yapılan işlemler ve işlemin girdileri hakkında bilgi sahibi olmaya çalışmaktadırlar [29, 50]. Şekil 3.6'da zamanlama analizi saldırısının kavramsal arayüzü gösterilmiştir.



Şekil 3.6: Zamanlama analizi saldırılarının kavramsal bir görünümü [50].

İşlemcide yürütülen farklı işlemler farklı sürelerde hesaplanmaktadır. Çarpma gibi bazı mantıksal işlemlerin yürütme süresi giriş değerine bağlı olarak değişebilmektedir. Kriptografik işlem gerçekleştiren donanımlarda şifreleme algoritmaları da değişken yürütme süreleriyle hesaplamalar yapılabilmektedir. Bu süre farklılıkları bir anahtara bağlı değişiyorsa değişen

yürütme süreleri analiz edilerek anahtar elde edilebilmektedir [50]. Zamanlama saldırısında sisteme yapılan farklı girişler için şifreleme işleminin yürütme süreleri analiz edilerek anahtar bitleri tahmin edilebilir. Bunun için öncelikle gecikme sinyallerini elde etmek gerekir. Sisteme farklı girişler uygulanarak sistem çıkışı güncellenmeden önce meydana gelen gecikmeler kaydedilerek sinyal gecikmeleri elde edilebilmektedir [29]. Örneğin parola kontrolünde çalışan bir sistem doğru parola girildiğinde yeşil ışık yanlış parola girildiğinde kırmızı ışık yaktığı durumda sisteme gireceğimiz parola tahminleri sonrası kırmızı ışığı tepki süresi arasında ilişki kurularak sistemin gerçek parolası elde edilebilir.

Zamanlama analizi genellikle güç analizi saldırıyla birlikte uygulanan bir saldırıdır. Elde edilen güç izi hem işlem hakkında bize bilgi verip güç analizi yapmamızı sağlarken hem de yapılan işlem tamamlanana kadar geçen süre hakkında bilgi vererek zamanlama analizi de yapmamızı sağlayabilmektedir [29].

Zamanlama analizi konusundaki ilk çalışmalardan birisini yapan Paul Kocher [51]'de yer alan çalışmasında Diffie-Hellman ve RSA için zamanlama saldırılarını göstermiş ve alınabilecek karşı önlemleri anlatmıştır. [52]'de araştırmacılar Qualcom Adreno 430 GPU (Graphics Processing Unit - Grafik İşlem Birimi) üzerinde çalışan AES şifreleme algoritmasının 128 bitlik anahtarını zamanlama yan kanal bilgisi ile elde ettikleri çalışmayı paylaşmışlardır. [53]'te yer alan çalışmada zamanlama saldırılarının tespiti ve zamanlama sızıntılarını azaltmak için teknikler analiz edilmiştir. [54]'te zamanlama saldırısının web sunucusuna uygulandığı çalışma paylaşılmış ve OpenSSL tabanlı bir web sunucusundan zamanlama saldırısı ile gizli anahtarın elde edilmesi anlatılmıştır. [50]'de yapılan çalışmada FPGA (Field-Programmable Gate Array) donanımı üzerinde zamanlama saldırılarına karşı güvenliğin artırılabileceği yöntem sunulmuş ve zamanlama saldırısı düzenlenerek yöntemin güvenlik seviyesine etkisi değerlendirilmiştir.

3.2. TEST ORTAMI

Tez çalışması kapsamında yan kanal analizi yapmak için hedef donanımlar ATmega328P ve ATXMEGA128D4-AU olarak seçilmiş ve hazırlanan senaryolara uygun yazılımlar hazırlanarak değerlendirme kartları aracılığıyla mikrodenetleyicilerin içerisine yüklenmiştir. Hedef donanımlara yan kanal saldırıları arasında zamanlama analizi ve güç analizi saldırıları uygulanarak güvenlik durumları değerlendirilmiştir. Aynı zamanda koruyucu önlem

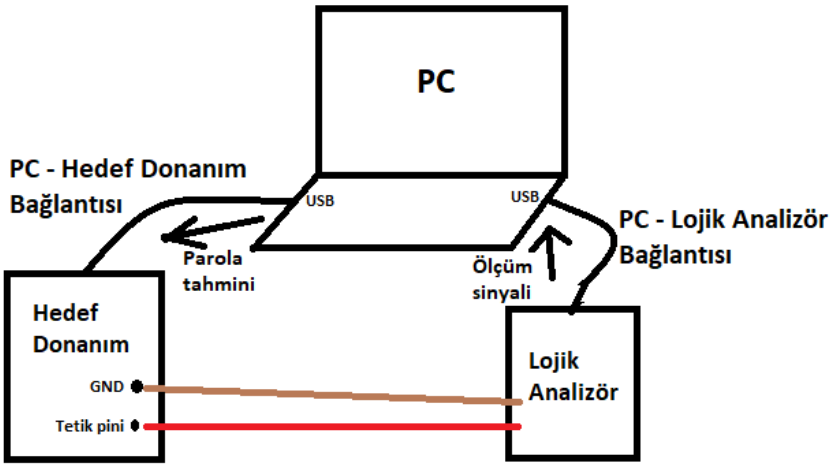
senaryoları uygulanmış ve bu senaryolara da yan kanal saldırıları düzenlenerek güvenlik durumları değerlendirilmiştir.

Donanımlar üzerinde yan kanal saldırılarının yapılabilmesi için donanımların zamanlama analizi için zamanlama çıkışı alınan pini üzerinden lojik analizör kullanılarak işlem süreleri ölçümü, güç sinyallerinin analizi için ise mikrodenetleyicilerin GND (Ground – Topraklama) bacaklarına direnç yerleştirilerek osiloskop üzerinden güç ölçümleri yapılmıştır.

Şifreleme çalıştıran donanım üzerinde basit güç analizi gerçekleştirmek ve güvenlik durumunu değerlendirmek için Chipwhisperer aracı kullanılmıştır.

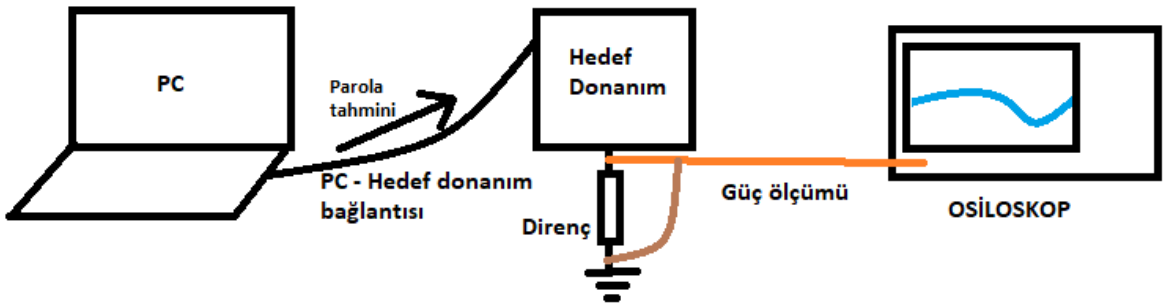
Zamanlama analizi saldırıları ile hedef donanım üzerinde çalışan parola doğrulama yapısına saldırı yapılmış ve sistem parolası elde edilmeye çalışılmıştır. Basit güç analizi saldırısında ise sistem içerisinde çalışan şifreleme algoritmasının anahtarının elde edilmesi amaçlanmıştır.

Hedef donanım üzerinde zamanlama analizi ile parolanın elde edilmesi için oluşturulan senaryoda ölçülen süre, klavyeden ‘enter’ tuşuna basılmasıyla ekrana uyarı mesajının çıkması arasında geçen süredir. Bunun için yazılım içerisinde bu iki noktayı belirten tetikleme mekanizması konulmuştur. Tetikleme mekanizması ‘enter’ tuşuna basıldığı donanım üzerindeki bir pini lojik 1 konumuna getirmektedir. Ekrana uyarı mesajı çıktığı anda ise lojik 0 durumuna geri dönmektedir. Bu şekilde tetik pini üzerinden lojik analizör aracı ile doğrulama işleminin süre ölçümü yapılmıştır.



Şekil 3.7: Lojik analizör ile hedef donanım üzerinden işlem süresi ölçümü için oluşturulan test ortamı.

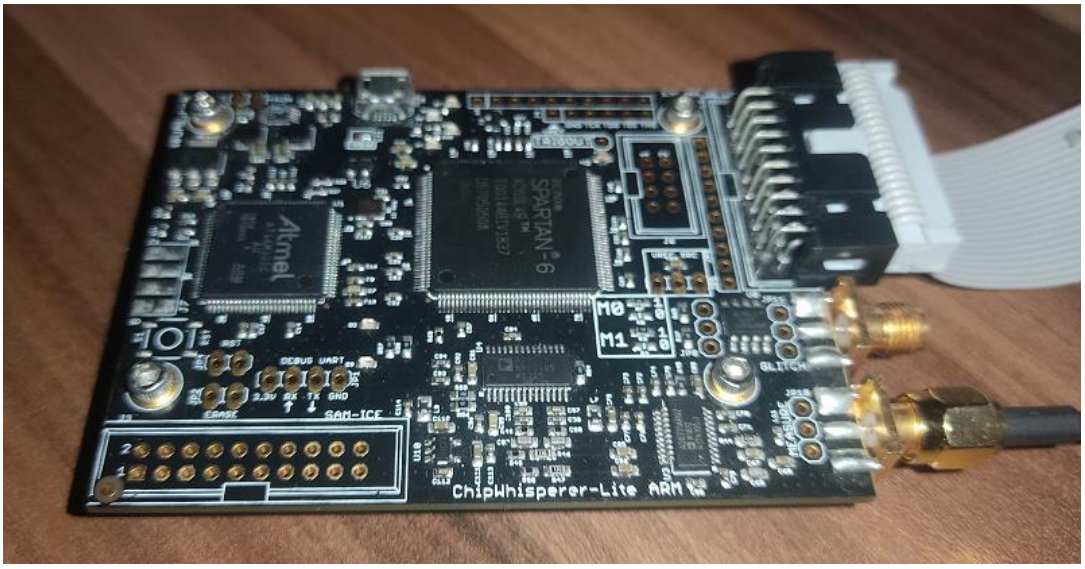
Zamanlama analizi saldırıları için ayrıca güç sinyalleri elde edilerek ve analiz edilerek de güvenlik değerlendirmesi yapılmıştır. Bu işlem için mikrodenetleyicinin topraklama bacağı geliştirme kartından sökülerek karttaki topraklama yolu ile arasına direnç koyulmuştur. Bu şekilde mikrodenetleyicinin anlık çektiği güç hakkında bilgi elde edilmiştir. Direnç üzerindeki gerilim osiloskop yardımıyla ölçülerek mikrodenetleyicinin istenilen andaki işlemlere ait güç izi elde edilmiştir.



Şekil 3.8: Mikrodenetleyicinin güç ölçümü için oluşturulan test ortamı.

3.2.1. Chipwhisperer

Tez çalışmasında yapılan deneylerden birisi olan basit güç analizi saldırısı için Chipwhisperer-Lite aracı kullanılmıştır. Chipwhiperer-Lite aracı gömülü sistem donanımlarındaki yan kanal saldırılarını öğretmek, kütüphanesinde bulunan donanımlara uygulamak için geliştirilmiş, geniş bir eğitim kütüphanesi bulunan açık kaynak yan kanal saldırı aracıdır. 105 MS/s ADC (Analog Digital Converter – Analog-Dijital Dönüştürücü) özelliğine sahip Chipwhisperer-Lite ile hedef donanımın güç izi yakalanarak analiz edilebilir, hedef donanım üzerinde gücü veya clock mekanizmasını kesintiye uğratarak hedef donanım üzerinde istenmeyen davranışlara neden olan saldırılar düzenlenebilmektedir. Araç, Python dili ile kullanılabilmektedir. Jupyter Notebook üzerinde çalışan araç yazılımı ile desteklenen donanımların yan kanal analizleri kolayca yapılabilmektedir [55].



Şekil 3.9: Tez çalışmasında kullanılan Chipwhisperer-Lite aracı.

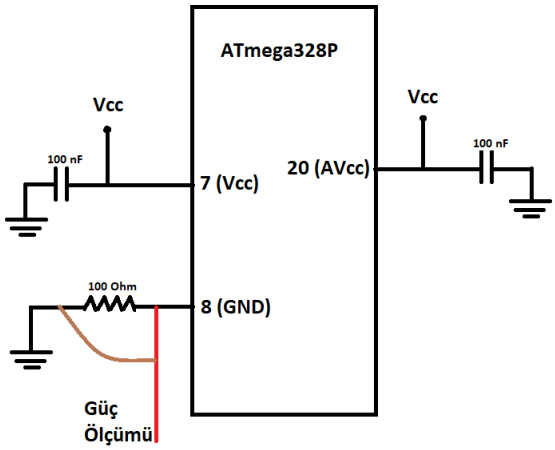
ATXMEGA128D4-AU mikrodenetleyicisi şifreleme yaparken mikrodenetleyiciye uygulanan basit güç analizi saldırısı Chipwhisperer-Lite aracı ile yapılmıştır.

3.2.2. ATmega328P Mikrodenetleyicisi

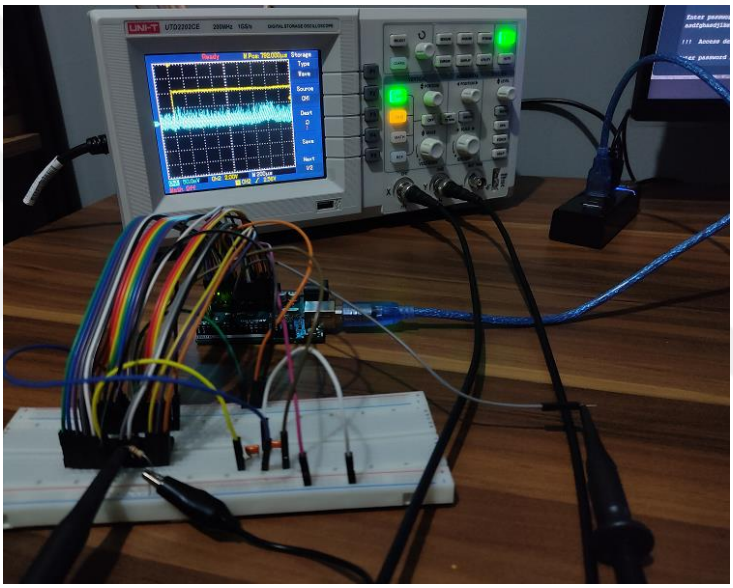
ATmega328P, hobi elektronik projelerinin sıklıkla kullanılan Arduino Uno R3 geliştirme kartlarında bulunan 8-bit AVR RISC tabanlı mikrodenetleyicidir [56]. ATmega328P mikrodenetleyicide 32 KB Flash hafıza, 2 KB SRAM (Static Random Access Memory – Durağan Rastgele Erişimli Bellek), 1024B EEPROM (Electronically Erasable Programmable Read-Only Memory - Elektrikle Silinip Programlanabilir Salt Okunur Bellek), 23 genel amaçlı I/O (Input/Output – Girdi/Çıktı) yolu bulunmaktadır [57].

ATmega328P mikrodenetleyicisi değerlendirmesi için yapılan tez çalışmasında Arduino UNO R3 kartı kullanılmıştır. Arduino programlama dili ile hazırlanan senaryolara uygun yazılımlar geliştirilerek Arduino Uno R3 kartına yüklenmiş, ATmega328P mikrodenetleyicisinin ve yazılımın güvenlik durumu incelenmiştir. Güç ölçümü için Arduino Uno R3 geliştirme kartından sökülerek bir ‘breadboard’ üzerine koyulmuş ve topraklama pinine 100 Ohm değerinde direnç bağlantısı yapılmıştır. Ayrıca güç sinyalini gürültülerden arındırmak için filtre devresi bağlantısı yapılmıştır. Böylece daha temiz bir sinyal elde edilmiştir.

Yazılım içerisinde Arduino Uno R3 geliştirme kartı üzerinde bulunan 7 numaralı dijital pine tetikleme yapılarak zamanlama analizi için işlem süresi ölçümü bu pin üzerinden yapılmıştır.



Şekil 3.10: ATmega328P mikrodenetleyicisi üzerinden güç ölçümü kurulumu.



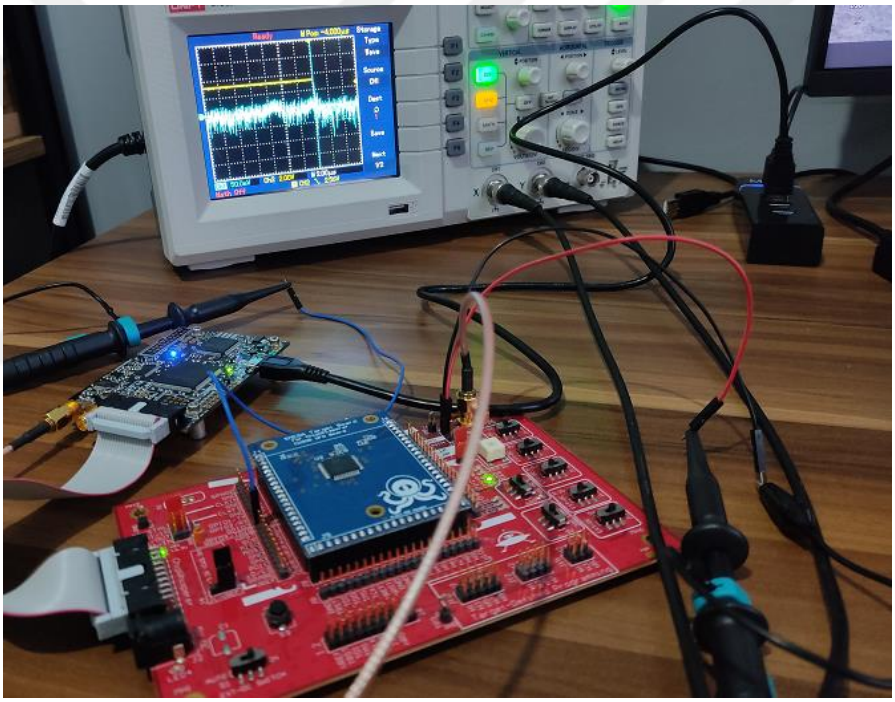
Şekil 3.11: ATmega328P mikrodenetleyicisinin güç analizi ve zamanlama analizi için kurulan test ortamı.

3.2.3. ATXMEGA128D4-AU Mikrodenetleyicisi

Tez çalışmasında güvenlik değerlendirmesi yapılan ikinci mikrodenetleyici ATXMEGA128D4-AU mikrodenetleyicisidir. ATXMEGA128D4-AU üzerinde hem parola doğrulamasına karşı zamanlama analizi hem de RSA şifreleme algoritmasına karşı basit güç analizi saldırısı yapılmıştır.

ATXMEGA128D4, yüksek performans ve düşük güç tüketimi sahip 8-bit AVR XMEGA mikrodenetleyicisidir. 128 KB 'Flash' hafızaya, 2 KB EEPROM ve 8 KB SRAM özelliklerine sahiptir. Üzerinde 34 tane genel amaçlı I/O yolu bulunmaktadır [58].

ATXMEGA128D4-AU mikrodenetleyicisinin testleri Chipwhisperer ürün ailesinde bulunan CW308 UFO kartı kullanılarak gerçekleştirilmiştir. CW308 UFO kartı üzerinde mikrodenetleyicinin pin çıkışları ve güç ölçümü için kurulan devre hazır olarak gelmektedir [59]. Chipwhisperer aracılığıyla doğrudan iletişim kurularak hem programlama işlemi hem de güç izi yakalama ve analiz işlemleri gerçekleştirilebilmektedir. Zamanlama analizi için tetikleme çıkışı verilen pin ayarlanarak lojik analizör ile işlem süresi elde edilmiştir. Güç izinin yakalanması için çıkış kart üzerinde çıkış verilen pinlere osilokop probu ile bağlantı yapılmış ve güç sinyali elde edilmiştir. Basit güç analizi için üzerinde ATXMEGA128D4-AU mikrodenetleyicisi bulunan CW308 UFO kartı doğrudan Chipwhisperer-Lite aracı ile kullanılmıştır. Senaryolar için oluşturulan yazılımlar C dili kullanarak yazılmıştır.



Şekil 3.12: ATXMEGA128D4-AU mikrodenetleyicisi için hazırlanan test ortamı.

3.2.4. Test Senaryoları

Hedef donanımlara ait yazılımların içerisinde bulunan parola doğrulaması ve şifreleme işlemi farklı senaryolar üzerinde değerlendirilmiştir. Her iki mikrodenetleyicide çalışan parola doğrulamasına zamanlama analizi saldırısı testleri için 4 farklı senaryo oluşturulmuştur. Bunlar:

- Parola kontrolünde açık metin olarak tutulan sistem parolasının karakterleri ile kullanıcının girdiği parolanın karakterlerinin teker teker karşılaştırılması

- Parola kontrolünde açık metin olarak tutulan sistem parolası ile kullanıcının girdiği parolanın kelime olarak karşılaştırılması

- Parola kontrolünde sistem parolasının MD5 (Message-Digest algorithm 5) hash değerinin tutularak kullanıcının girdiği parolanın MD5 hash değeriyle karşılaştırılması

- Parola kontrolü işleminin öncesinde ve sonrası rastgele zaman gecikmesi eklenmesi

Üzerinde RSA şifreleme işlemi gerçekleşen ATXMEGA128D4-AU mikrodenetleyicisine ayrıca basit güç analizi saldırısı düzenlenmiş ve şifreleme adımları arasına mikrodenetleyicide güç harcanmasına sebep olacak anlamsız ve mikrodenetleyici üzerinde çalışan yazılıma etki etmeyen kod parçacıkları eklenerek basit güç analizi saldırısı yinelenerek güvenlik durumları değerlendirilmiştir.

4. BULGULAR

4.1. PAROLA KORUMALI SİSTEME ZAMANLAMA ANALİZİ SALDIRISI

Yetkisiz sistem erişimini önlemek amacıyla donanım içerisindeki yazılımda bulunan parolaları elde edebilmek amacıyla sistemin parola kontrol ekranı açılmıştır. Girilen tahminlerin zaman gecikmesi için parolanın yazılıp klavyeden ‘enter’ tuşuna basıldıktan sonra ekranda çıkan ‘Erişim engellendi.’ uyarısına kadar geçen süre ölçülmüş ve girilen tahminlerin yanlış parola hatası çıkana kadar ki süreleri analiz edilerek sisteme ait parolaya ulaşılmıştır. Bu test senaryosundaki düşünce girilen hatalı karakterlerin aynı gecikmeye ve güç profiline sahip olacağı, doğru karakterin ise farklı bir çıktı vermesidir. Aynı zamanda bu açığı kapatmak için parolanın sistem kodlarında hash değerinin tutulması ve kod içerisinde parola kontrolü aşamasında rastgele zaman gecikmesi eklenmesi senaryolarının güvenlik durumları değerlendirilmiştir.

4.1.1. ATmega328P Mikrodenetleyici Üzerinde Parolanın Karakter Olarak Karşılaştırılması

Üzerinde ATmega 328P mikrodenetleyici bulunan Arduino Uno geliştirme kartı ile hazırlanan parola korumalı sistemde parolanın karakter karakter karşılaştırılarak parola kontrolü yapılan sistem hazırlanmıştır.

```
for (int i = 0; i < systemPassword.length(); i++) {
  if (password[i] != systemPassword[i]) {
    passFlag = 1;
    break;
  }
}
```

Şekil 4.1: Arduino içerisinde yapılan parolanın karakter olarak karşılaştırılması işlemi.

Parola seri port üzerinden kullanıcıdan istenmektedir. Girilen parola sonrasında ekrana çıkan hata uyarısı arasındaki süre ölçülerek analiz işlemi yapılmış ve parola elde edilmiştir.

```
Enter password for system access:

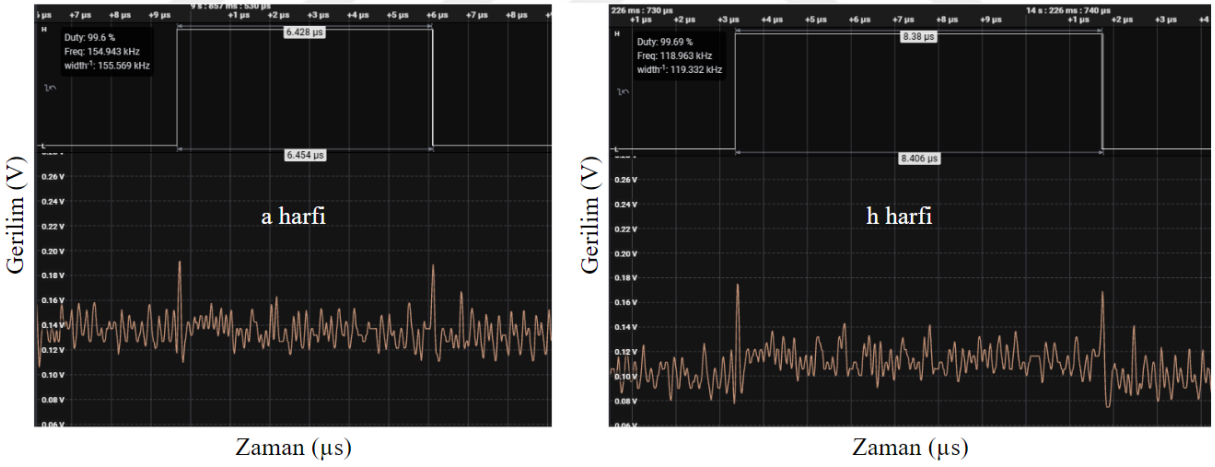
a

Access denied. Password is wrong.
```

Şekil 4.2: Seri port üzerinden girilen hatalı parola sonrası karşılaşılan hata uyarısı.

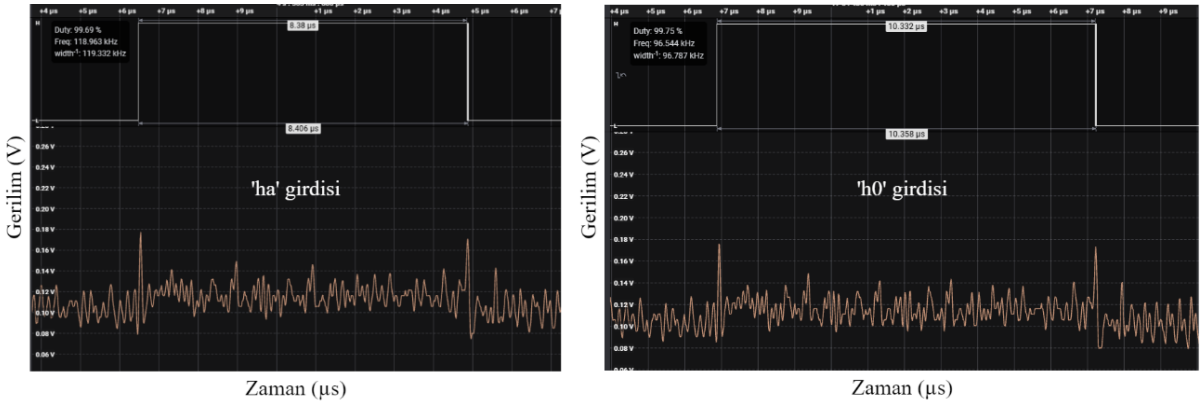
Uyarı mesajının ‘enter’ tuşuna basıldıktan sonra ekranda belirme süresinin ölçülmesi için kod içerisinde tetik sinyali oluşturulmuş ve bu sinyal üzerinden süre ölçümü yapılmıştır. Tetikleme mekanizması ‘enter’ tuşuna basıldıktan sonra Arduino Uno üzerindeki 7 numaralı dijital pini 5V seviyesine çekmekte ve ekranda uyarı yazısı çıktığı anda 0V seviyesine indirmektedir.

İlk olarak tek tek harfler denenmiş ve girilen karakterlerin zaman gecikmeleri incelenmiştir. Girilen tüm hatalı harfler birbiriyle aynı veya çok yakın zaman gecikmesi oluştururken, ‘h’ harfi daha uzun süren bir zaman gecikmesine sebep olmuştur. Bu durumla birlikte parolamızın ilk harfini elde etmiş oluyoruz. Şekil 4.3 ‘a’ ve ‘h’ harflerinin lojik analizör ile ölçülen zaman gecikmelerini göstermektedir.



Şekil 4.3: ‘a’ ve ‘h’ harflerinin zaman gecikmesi

Parolanın ilk harfi elde edildikten sonra ‘h’ harfini sabit tutup yanına ikinci karakterleri ekleyip parola denemesi yapılarak parolanın ikinci harfinin elde edilmesi işlemi gerçekleştirilir. İlk karakterde olduğu gibi hatalı tüm karakterlerde aynı veya yakın zaman gecikmesi elde edilirken ‘h0’ denemesinde daha uzun bir zaman gecikmesi elde edilmektedir. Bu şekilde parolanın ikinci karakteri de elde edilmiştir. Şekil 4.4 ‘ha’ ve ‘h0’ parola denemeleri için zaman gecikmelerini göstermektedir.



Şekil 4.4: ‘ha’ ve ‘h0’ girdilerinin zaman gecikmesi.

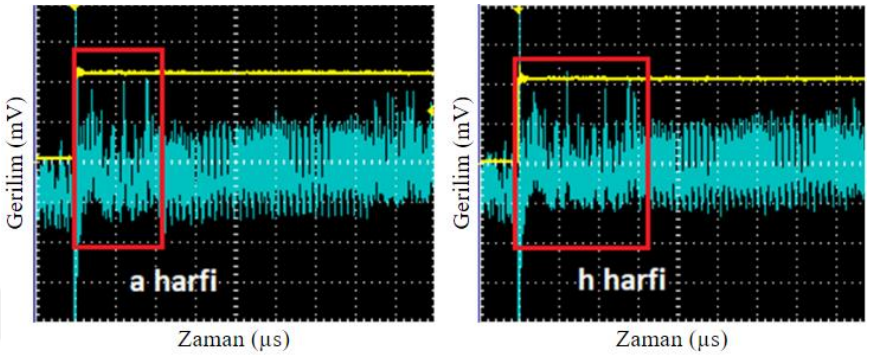
Bu şekilde her adımda bir sonraki doğru karakter tespit edilerek sistemin parolası olan ‘h0pe’ elde edilmiştir. Girilen karakterler ve elde edilen zaman gecikmelerinden bazıları Tablo 4.1’de verilmiştir.

Tablo 4.1: Seri port üzerinden yapılan girdiler ve lojik analizör ile ölçülen zaman gecikmeleri

Girilen Karakter	Süre(µs)
\n	6,428
a	6,428
aa	6,428
aaa	6,428
aaaa	6,428
aaaaa	6,428
b	6,426
2	6,428
h	8,380
ha	8,380
hh	8,380
h1	8,380
h0	10,332
h0a	10,332
h00	10,332
h0p	12,284
h0pa	12,284
h0pp	12,284
h0pe	sistem erişimi

Sistemin parola kontrolüne ait zaman analizi lojik analizör ile ‘enter’ tuşunun tıklanması ve ekranda uyarı yazısının çıkması arasındaki süre ölçülerek yapılabileceği gibi

mikrodenetleyicinin güç tüketimi ölçülerek de yapılabilir. Arduino'da bulunan Atmega mikrodenetleyicisinin osiloskop ile harcadığı güç çıkarılarak, güç sinyallerindeki gecikmelerin değerlendirilmesiyle sistem parolası elde edilebilmektedir. Bunun için yine klavyeden 'enter' tuşuna basıldıktan sonraki belirli bir sürede sistemin harcadığı güç sinyali elde edilmiş ve sinyaldeki gecikmeler analiz edilerek parolaya erişilmiştir. Şekil 4.5'te 'a' ve 'h' harflerinin karşılaştırılması sırasında elde edilen güç profilleri gösterilmektedir.



Şekil 4.5: 'a' ve 'h' harflerinin işlem esnasında harcadıkları gücü gösteren mikrodenetleyiciye ait güç sinyallerinin osiloskop görüntüleri.

Şekil 4.5'teki güç profilleri incelendiğinde kırmızı çerçeve içerisine alınmış işlemin genişlediği görülmektedir. Lojik analizör ile yapılan karakter karakter parolanın elde edilmesi işlemi osiloskop kullanılarak her adımda bir karakter bulunarak sistemin parolası elde edilmiştir.

4.1.2. ATmega328P Mikrodenetleyici Üzerinde Parolanın Kelime Olarak Karşılaştırılması

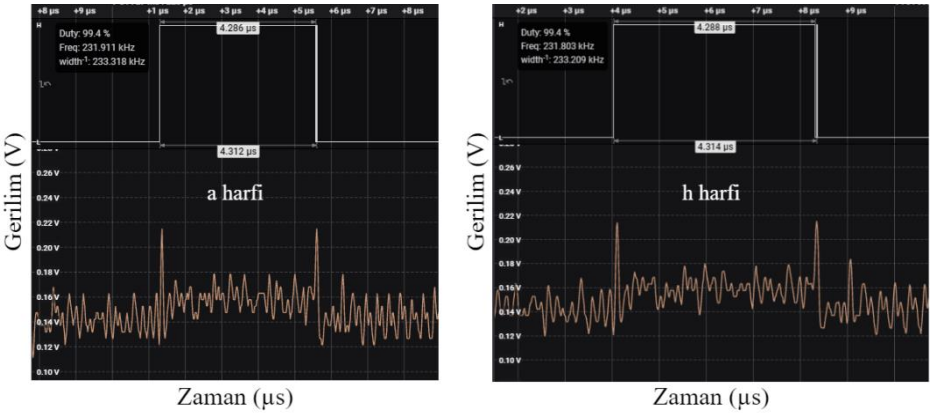
Bölüm 4.1.1'de Arduino Uno geliştirme kartı ile ATmega328P üzerinde parola kontrolü, kullanıcı tarafından girilen parolanın karakterleri ile sistem parolasının karakterleri ilk karakterden başlayarak karakter karakter karşılaştırılarak yapılmıştı. Bu senaryoda daha yaygın kullanılan parolanın kelime olarak karşılaştırılması işlemi yapılmış ve zaman gecikmesi değerlendirilmiştir.

```
if (password == systemPassword) {
  Serial.println("System access successful.\n");
```

Şekil 4.6: Arduino içerisinde yapılan parolanın kelime olarak karşılaştırılması işlemi.

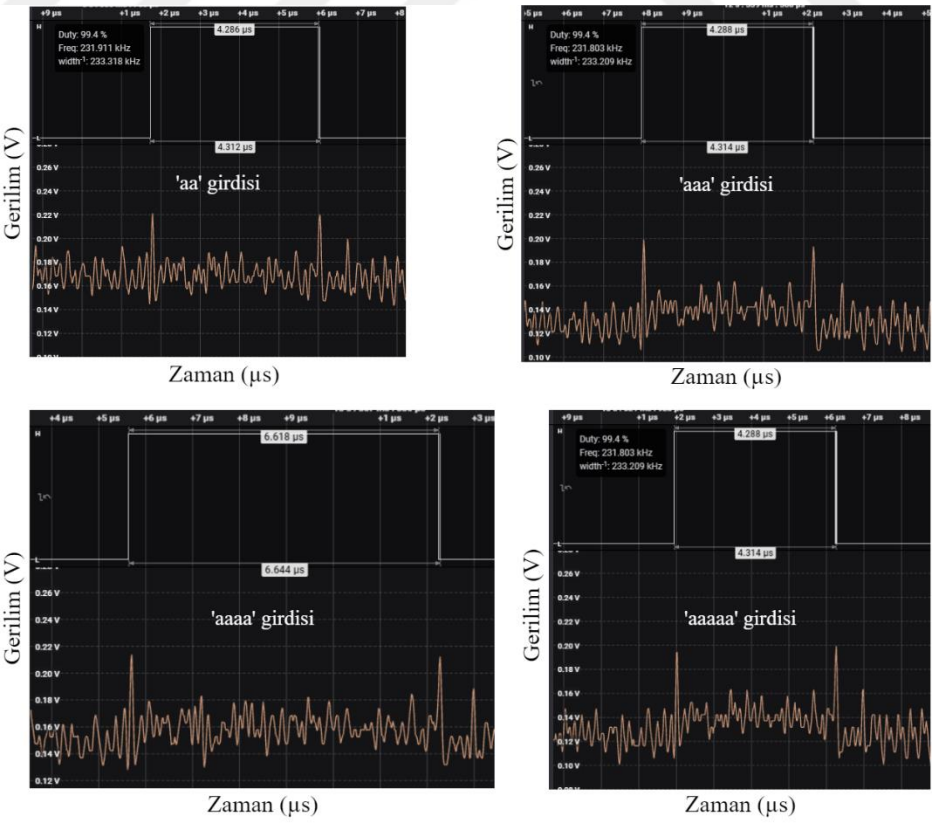
Lojik analizör ile yapılan zaman gecikmesi için uygulanan senaryo Bölüm 4.1.1'de anlatıldığı gibi 'enter' tuşuna basıldıktan sonra ekrana hata uyarısı çıkana kadar geçen süre ölçülmüştür.

Seri port üzerinden ilk olarak harf ve rakam denemeleri yapılmış ve karakter bazlı bir zaman gecikmesi gözlenmemiştir.



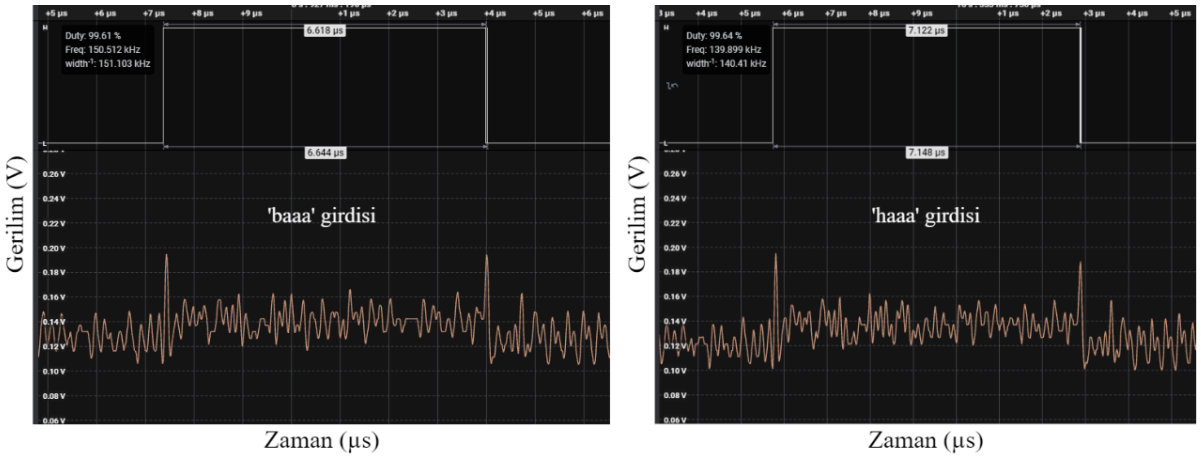
Şekil 4.7: 'a' harfi ve 'h' harfine ait zaman gecikmesi.

Karakter olarak işlem sürelerinde bir gecikme gözlenmemesinin ardından kelime dizisi olarak girdiler yapılmış ve parola hakkında bilgi elde edilmeye çalışılmıştır. Bunun için ‘aa’, ‘aaa’, ‘aaaa’ ve ‘aaaaa’ girdileri yapılmış ve işlem süreleri incelenmiştir. Buradaki amaç parola uzunluğu hakkında mikrodenetleyici üzerinden bilgi alabilmektir.



Şekil 4.8: 'aa', 'aaa', 'aaaa' ve 'aaaaa' girdileri sonrası ölçülen işlem süreleri.

Şekil 4.8’de, yapılan girdiler sonrası elde edilen işlem süreleri incelendiğinde ‘aaaa’ girdisinin diğer girdilere göre daha uzun işlem süresine sahip olduğu görülmektedir. Bu durum bize parolamızın 4 haneli olduğu bilgisini vermektedir. Parolanın kaç karakterden oluştuğu öğrenildikten sonraki aşamada sisteme 4 haneli tahminler iletilerek parola hakkında daha fazla bilgi edinilmeye çalışılır. Bunun için ‘aaaa’ girdisi sonrasın ilk karakter değiştirilerek (‘baaa’, ‘caaa’, ‘daaa’ gibi) denemeler yapılmıştır. Yapılan denemelerin işlem süreleri incelendiğinde ‘haaa’ girdisinin işlem süresinin diğer girdilerden daha uzun olduğu görülmüştür.



Şekil 4.9: 'baaa' ve 'haaa' girdileri sonrası ölçülen işlem süreleri.

Bu test senaryosunda karakter sayısı sabit tutulup her adımda o adıma ait parola karakteri elde edilmiştir. ‘haaa’ girdisinin işlem süresinin diğer girdilerin işlem süresinden fazla olması sebebiyle 4 karakterden oluşan parolamızın ilk karakterinin ‘h’ olduğunu sonucuna erişmekteyiz. Bir sonraki adımda tespit edilen ilk karakteri sabit tutup bir sonraki karakter değiştirilerek (‘hbaa’, ‘hcaa’ gibi) işlem süreleri ölçülmüş ve ‘h0aa’ girdisi hariç diğer girdilerin işlem süreleri aynı veya yakın değerler çıkmıştır. Böylelikle sistem parolasının ikinci harfinin ‘0’ olduğu anlaşılmıştır. Aynı şekilde bir sonraki karakteri elde etmek için elde edilen ilk iki karakter sabit tutularak üçüncü karakter değiştirilmiş (‘h0ba’, h0ca’ gibi) ve işlem süreleri incelendiğinde ‘h0pa’ girdisinde işlem süresinin daha uzun olduğu görülmüştür. Bu sonuçla birlikte parolaya ait üçüncü karakterin ‘p’ harfi olduğu elde edilmiştir. Bir sonraki adımımız parolanın son harfini elde etmek için önceki adımlarda elde ettiğimiz ilk üç karakter sabit tutulup son karakter değiştirilerek (‘h0pa’, ‘h0pb’ gibi) son karakter ve dolayısıyla parolanın tamamı elde edilmeye çalışılmıştır. Bu noktada son karakteri aradığımız için işlem sürelerinin incelenmesi gerekmemektedir. Son karakter doğru olduğunda sistem erişimi sağlanacağından parola da elde edilmektedir. Yapılan denemelerde sistem erişimi sağlayan harf kombinasyonu

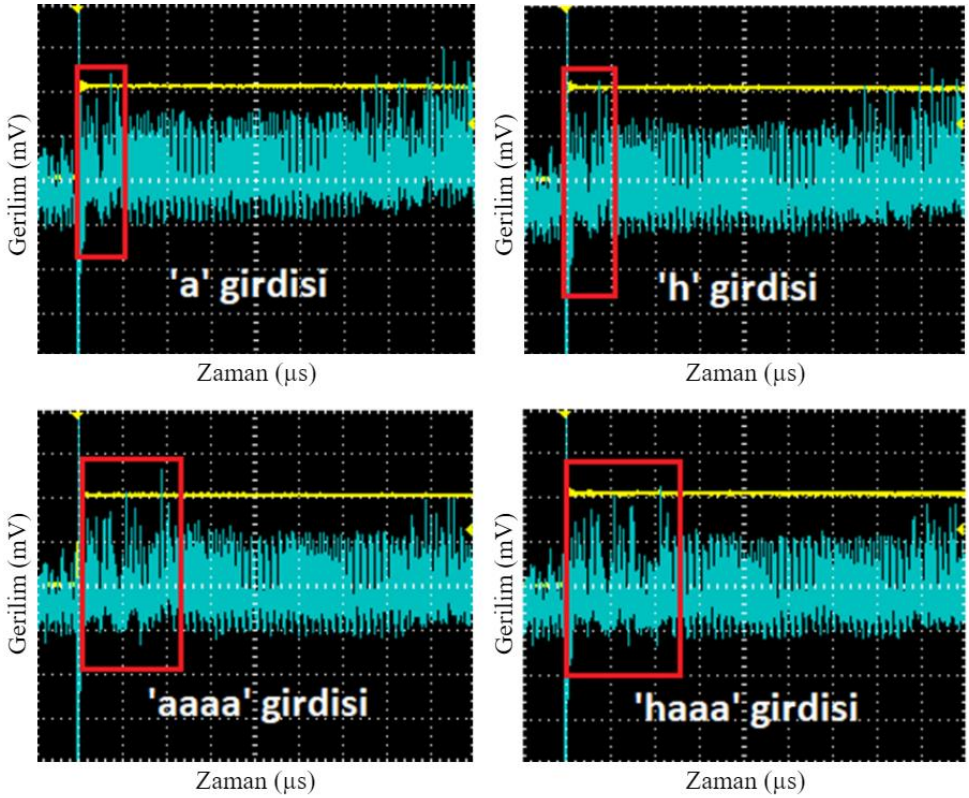
sisteme ait parola olmaktadır. Böylelikle bu işlem için asıl amaç olan sistem erişimi sağlanmaktadır. Bu şekilde adım adım ilerleyerek sistemin parolası olan ‘h0pe’ kelimesi elde edilmiştir.

Analiz sırasında mikrodenetleyiciye yapılan girdiler ve lojik analizör ile ölçülen işlem sürelerinden bazıları Tablo 4.2’de verilmiştir.

Tablo 4.2: Seri port üzerinden yapılan girdiler ve işlem süreleri.

Yapılan girdi	Ölçülen süre (µs)
\n	4,288
a	4,286
5	4,286
h	4,288
aa	4,286
aaa	4,288
aaaa	6,618
aaaaa	4,288
baaa	6,618
0aaa	6,618
haaa	7,122
hhaa	7,122
h1aa	7,120
h0aa	7,624
h00a	7,624
h0ba	7,624
h0pa	8,128
h0pp	8,128
h0p3	8,128
h0pe	sistem erişimi

Test çalışmasında lojik analizör ile ‘enter’ tuşuna basılmasıyla ekranda uyarı mesajı çıkması arasındaki süre ölçülerek işlem süreleri elde edilmiştir. Bu işlem için Bölüm 4.1.1’de bahsedildiği gibi osiloskop yardımıyla Atmega mikrodenetleyicisinin yapılan girdiler için harcadığı güç profili elde edilip güç izindeki gecikmeler analiz edilerek de parolanın tamamı elde edilebilmektedir.



Şekil 4.10: 'a', 'h', 'aaaa' ve 'haaa' girdilerinin işlem esnasında harcadıkları gücü gösteren mikrodenetleyiciye ait güç sinyallerinin osiloskop görüntüleri.

Şekil 4.10'da kırmızı dikdörtgen içerisine alınan nikrodenetleyiciye ait güç sinyalleri incelendiğinde 'a' ve 'h' girdilerinin benzer güç profiline sahip olduğu, 'aaaa' girdisi yapıldığında güç izinde genişleme olduğu görülmektedir. 'haaa' girdisi yapıldığında işleme ait sinyalin biraz daha genişlemesi bize parolanın ilk hanesinin 'h' olması bilgisini vermektedir.

4.1.3. ATmega328P Mikrodenetleyici Üzerinde Parolanın Hash Değerinin Karşılaştırılması

Daha güvenli parola korumalı bir sistem için parolalar bir hash fonksiyonu ile hash değerleri alınıp bu değerleri sistemde tutulabilir. Bu bölümdeki senaryoda sistem içerisinde kullanıcının yaptığı girdilerin MD5 hash değerleri alınıp sistemin parolasına ait MD5 hash değeriyle karşılaştırılarak işlem yapılmaktadır.

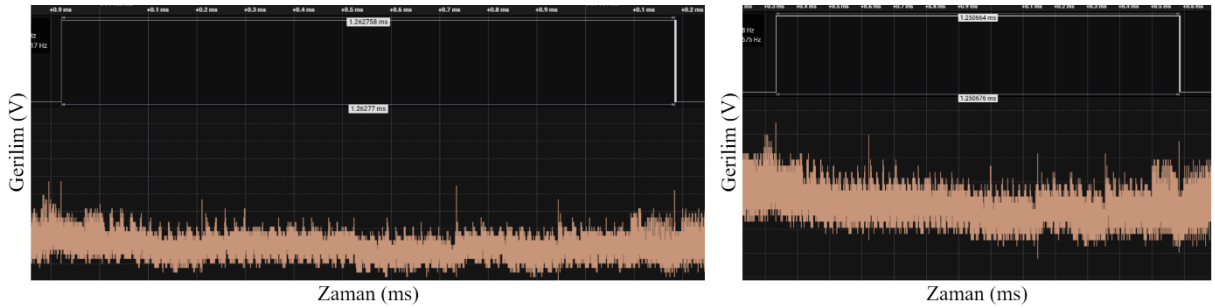
```
char *md5str = hashMD5.md5(plaintext_array);

if (strcmp(md5str, systemPasswordHash)==0) {
  Serial.println("\nSystem access successful \n");
```

Şekil 4.11: Açık metin halinde girilen parolanın hash değerinin alınıp sistemin hash değeriyle karşılaştırılması.

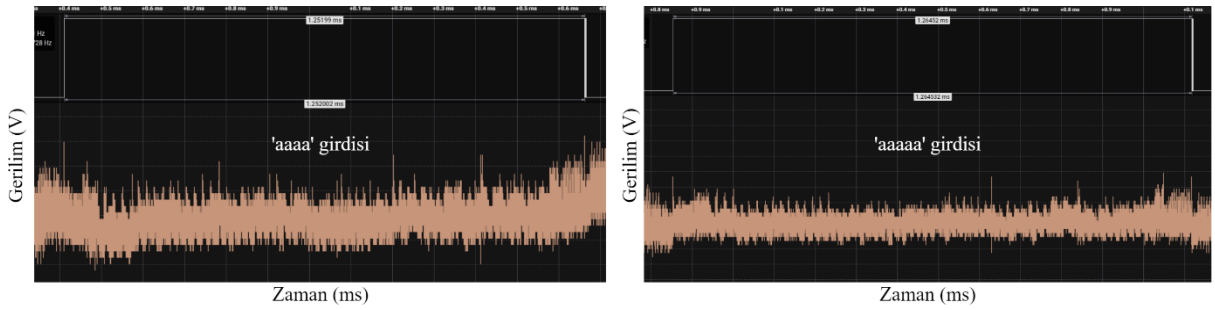
İşlem süresi ölçümü için hazırlanan senaryo Bölüm 4.1.1 ve Bölüm 4.1.2'de anlatılan senaryonun aynısıdır. Kullanıcının parolayı girip 'enter' tuşuna basmasıyla ekranda uyarı mesajının çıkması arasındaki süre ölçülerek zamanlama gecikmeleri analiz edilmiştir.

İlk olarak karakter denemeleri yapılmış ve karakterlerin hepsi farklılık göstermiş diğerlerinden ayrılan bir karakter ile karşılaşılmamıştır. Aynı karakter ikinci defa girildiğinde dahi farklı sonuç alınmıştır.



Şekil 4.12: 'b' girdisinin ilk süre ölçümü (solda) ve ikinci süre ölçümü (sağda).

Sisteme karakter sayıları artırılarak kelime girdileri yapılmış ve elde edilen süre ölçümleri arasında parola tahmini yürütülecek bir ilişki tespit edilememiştir.



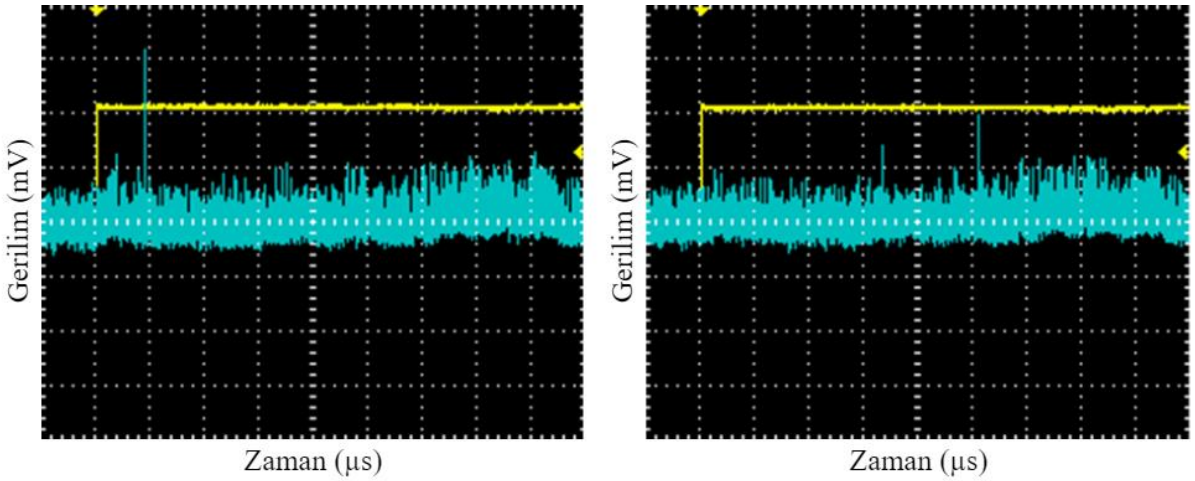
Şekil 4.13: 'aaaa' ve 'aaaaa' girdilerinin işlem süreleri.

Hem karakter olarak hem de karakter sayısı artırılarak kelime olarak birçok girdi yapılmıştır. Bazı girdiler tekrar yapılarak aynı girdiye ait farklı sonuçlar elde edildiği görülmüştür. Elde edilen işlem süreleri incelendiğinde parola hakkında tahminde bulunacak bilgiye erişilememiştir. İşlem sürelerinin Bölüm 4.1.1 ve Bölüm 4.1.2'de yapılan incelemelerde elde edilen mikrosaniye seviyesinde değil milisaniye seviyesinde olduğu görülmüştür. Yapılan girdiler ve girdilerin işlem süreleri Tablo 4.3'te verilmiştir.

Tablo 4.3: Seri port üzerinden yapılan girdiler ve girdilere ait işlem süreleri

Yapılan girdi	Süre(ms)
\n	1,247774
a	1,250666
aaa	1,257592
ddd	1,248532
aaaa	1,261054
eeee	1,251990
aaaaa	1,264520
b	1,262758
h	1,262756
cdf	1,263646
mxw	1,263642
prswqa	1,268494
baaa	1,264078
haaa	1,264086
hhaa	1,264088
h0aa	1,264078
h0pa	1,264080
h0pp	1,264080

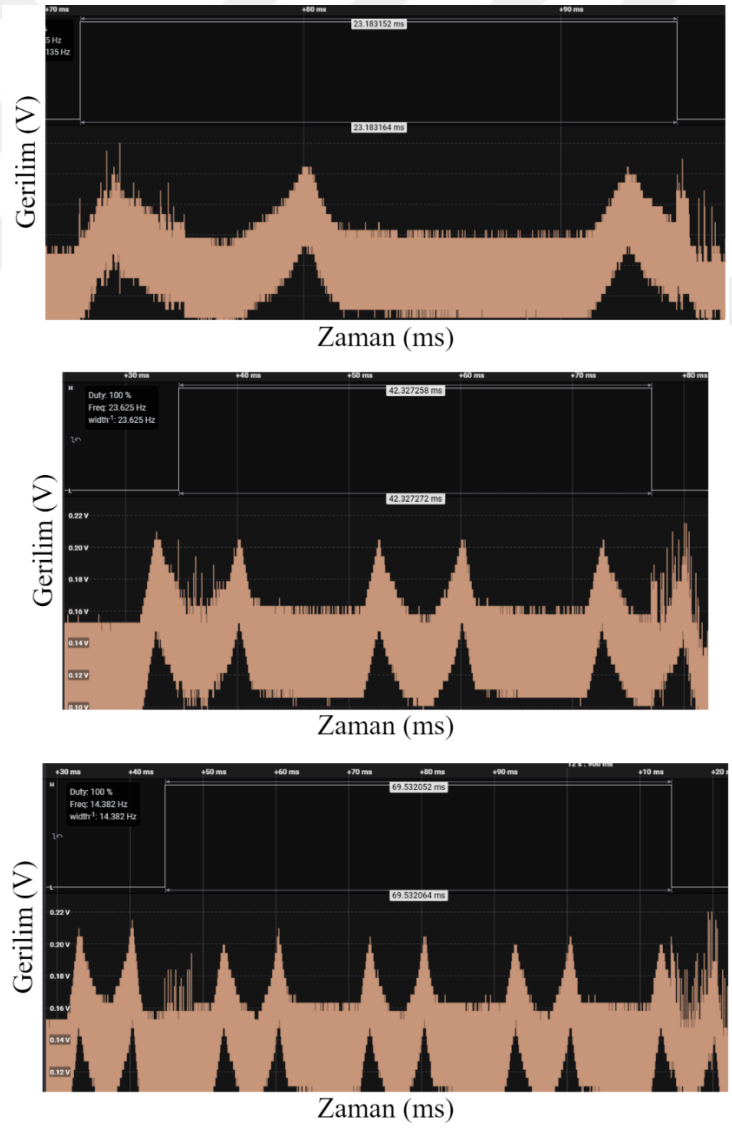
Mikrodenetleyicinin girilen girdiler sonrası yaptığı işlemlere ait güç sinyalleri osiloskop yardımıyla incelendiğinde parola hakkında bir bilgi tespit edilememiştir.



Şekil 4.14: 'a' girdisi ile sisteme yapılan iki denemeye ait güç sinyali

4.1.4. ATmega328P Mikrodenetleyici Üzerinde Parola Karşılaştırma Yapısına Rastgele Zaman Gecikmesi Koyularak Parolanın Kontrol Edilmesi

Bölüm 4.1.1 ve Bölüm 4.1.2'de içerisinde açık metin halinde parola tutan ve kullanıcının girdiği parola ile karşılaştıran sistemde karşılaştırma adımının işlem süresi ölçülerek hatalı tahminlerde benzerlik yakalanmış ve doğru parola karakteri için işlem süresinin uzun olduğu görülmüştür. Bu senaryoda işlem süreleri ve girilen karakterler arasındaki benzerliği ortadan kaldırmak için parola karşılaştırma yapısına rastgele zaman bekletmesi eklenmiş ve parola hakkında bilgi edinilmeye çalışılmıştır. Girilen parola tahminlerinin işlem süresi 'enter' tuşuna basıldıktan sonra ekranda uyarı mesajı çıkana kadar geçen süre ölçülerek elde edilmiştir. Sistem içerisinde parola karşılaştırma yapısında rastgele zaman gecikmesi oluşturulduğundan aynı karakter sisteme verildiğinde dahi farklı işlem süreleri elde edilmiştir.



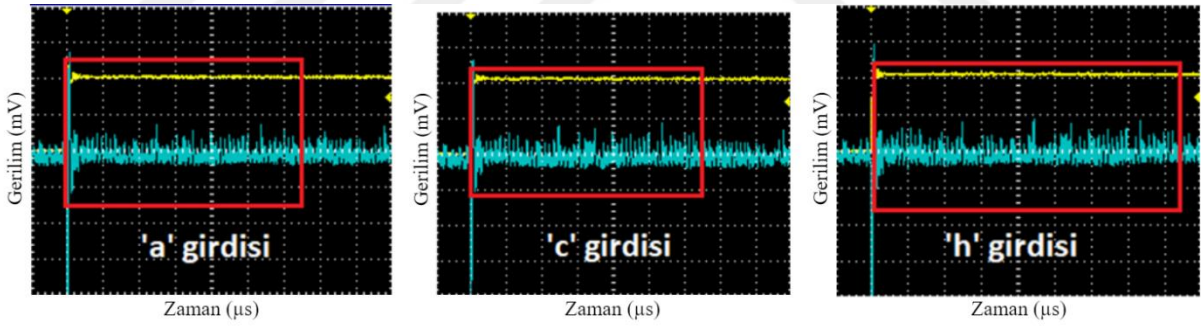
Şekil 4.15: Sisteme 3 defa 'h' girdisi yapıldığında elde edilen 3 farklı işlem süresi.

Lojik analizör ile ölçülen ‘enter’ tuşu ile uyarı mesajı arasındaki süreler incelendiğinde parola hakkında bilgi edinilememiştir.

Tablo 4.4: Lojik analizör ile farklı girdilerin 3 ayrı ölçümünden elde edilen işlem süreleri.

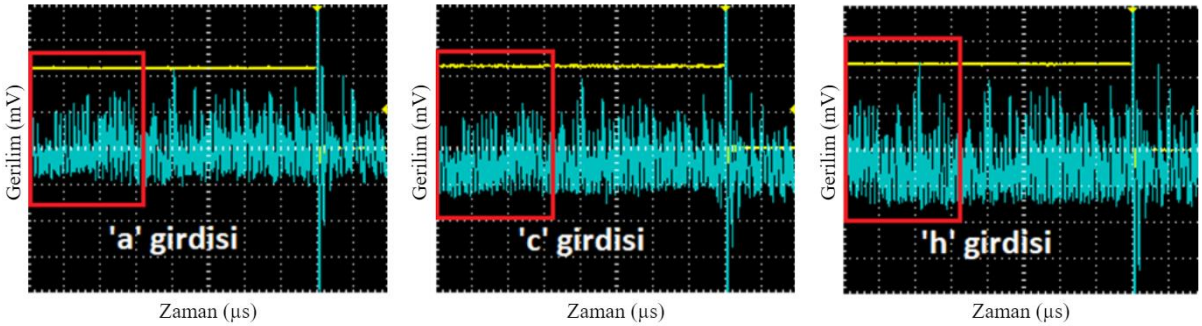
Girilen Karakter	1. ölçüm (ms)	2. ölçüm (ms)	3. ölçüm (ms)
\n	40,313908	28,221824	7,061494
a	99,759492	32,251182	33,261778
h	23,183152	42,327258	69,532052

Yapılan girdiler sonrası mikrodenetleyicinin harcadığı güç miktarı osiloskop yardımıyla incelendiğinde ise lojik analizör ile işlem süresi analizinin aksine parola hakkında bilgiye erişilebildiği görülmüştür. Şekil 4.16’da görüleceği üzere yanlış karakter benzer kalıpta güç profili ile başlarken doğru olan karakter farklı bir kalıp çizmiştir. Burada sistemin rastgele bekletme süresini parola karşılaştırma işleminden sonrasına koyulduğu bu sebeple ‘enter’ tuşu ardından elde edilen sinyalin başlangıç aşamasında parola hakkında bilgi yer aldığı sonucu çıkarılmıştır.



Şekil 4.16: 'a', 'c' ve 'h' girdisi için elde edilen mikrodenetleyicinin güç profilleri.

Şekil 4.17’de parola kontrolü öncesinde rastgele süre bekletme işlemi yapılmış sisteme ait mikrodenetleyicinin farklı girdiler sonucu elde edilen güç profili gösterilmektedir. Burada işlemin sonu ekrana uyarı yazısının çıkması anıdır. Güç profili incelendiğinde hatalı karakterlerin benzer güç profillerinin olduğu doğru karakter için elde edilen güç profilinin farklı olduğu tespit edilmiştir.



Şekil 4.17: 'a', 'c' ve 'h' girdileri için elde edilen mikrodenetleyicinin güç profilleri

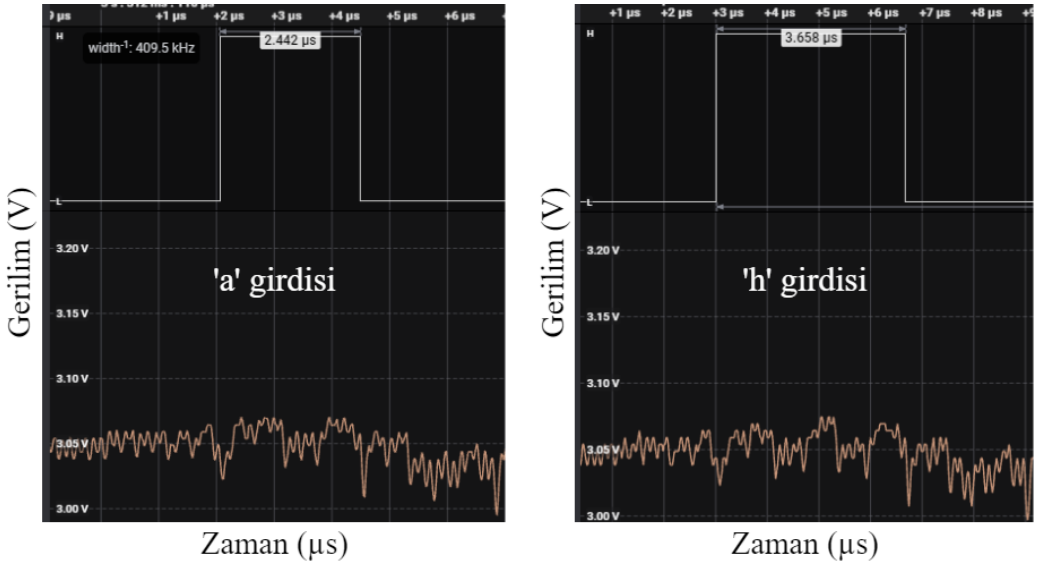
4.1.5. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parolanın Karakter Olarak Karşılaştırılması

XMEGA mikrodenetleyicisi üzerinde geliştirilen parola korumalı sistem erişimi uygulamasında parola kontrolü kullanıcının girdiği parolanın karakterleri ile sistemin parolasının karakterlerinin sırasıyla karşılaştırılmasıyla oluşturulmuş ve yapılan girdiler karşısında sistemin girilen parola karşısında işlem süreleri incelenmiştir.

```
for(uint8_t i = 0; i < sizeof(systemPassword); i++){
    if (systemPassword[i] != password[i]){
        passFlag = 1;
        break;
    }
}
```

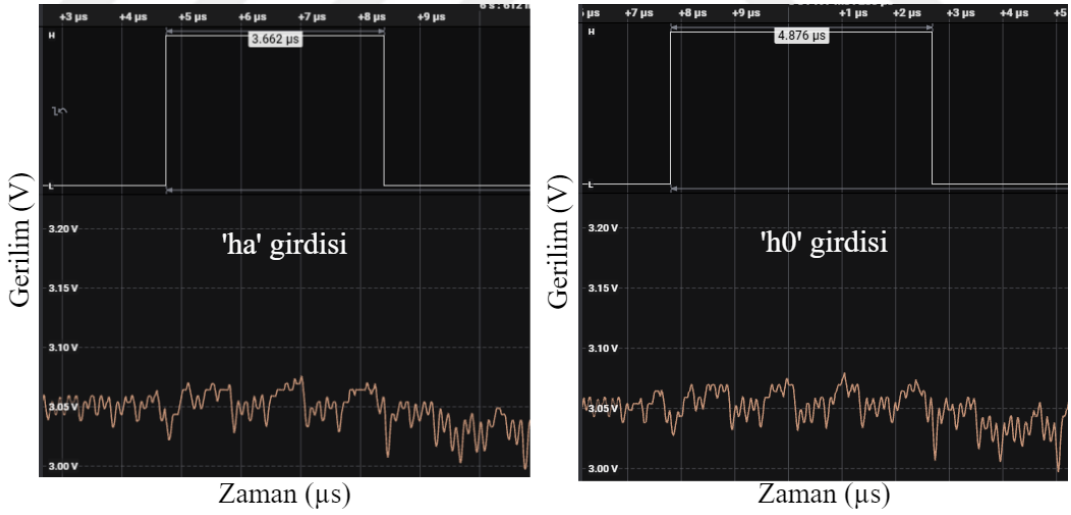
Şekil 4.18: XMEGA üzerinde yapılan parolanın karakter karakter karşılaştırılarak doğrulama yapılması.

Klavyeden ‘enter’ tuşuna basılmasıyla ekranda uyarı mesajı çıkması arasındaki işlem süresi ölçülerek zaman bilgileri analiz edilmiştir. Bu işlem için sistem içerisinde tetikleme mekanizması oluşturulmuş ve tetikleme kodu ‘enter tuşuna basıldığında XMEGA128D4-AU mikrodenetleyicisinin değerlendirilmesi için kullanılan CW308 UFO Kartı üzerinde GPIO4/TRIG pinini ‘HIGH’ (lojik 1) durumuna çekmiştir. Aynı şekilde ekranda uyarı mesajı çıktığı anda aynı pini ‘LOW’ (lojik 0) durumuna çekmiştir. Lojik analizör kullanılarak yapılan ölçümde öncelikle sisteme karakter girdileri yapılmış ve işlem süreleri incelenmiştir. Yapılan incelemede ‘h’ girdisine ait işlem süresinin diğer girdilerden daha uzun sürdüğü gözlenmiştir. Diğer girdilerin işlem süreleri aynı veya yakın olduğu görülmüştür. Yapılan bu ölçümle birlikte sisteme ait parolanın ilk harfinin ‘h’ olduğu tespit edilmiştir.



Şekil 4.19: 'a' ve 'h' girdileri sonrası mikrodenetleyici üzerinden lojik analizör ile ölçülen işlem süreleri.

İlk harf tespiti yapıldıktan sonra ilk harfi sabit tutup yanına farklı karakterler ekleyerek (‘ha’, ‘hb’, ‘hc’ gibi) parola elde etme işlemine devam edilmiştir. Bu adımda ‘h0’ girdisi dışındaki girdilerin işlem sürelerinin aynı veya benzer olduğu ‘h0’ girdisinin işlem süresinin diğer girdilere göre daha uzun olduğu görülmüştür.



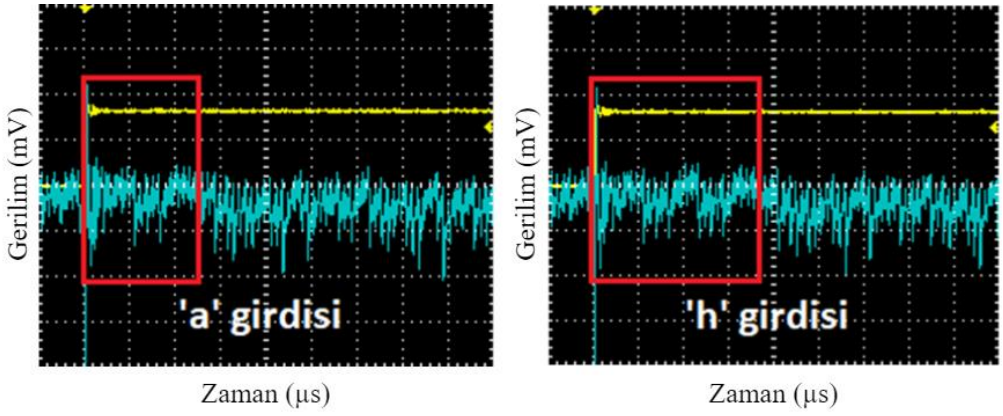
Şekil 4.20: 'ha' ve 'h0' girdileri sonrası mikrodenetleyici üzerinde lojik analizör ile elde edilen işlem süreleri.

İkinci harf tespit edildikten sonra işleme aynı adımlarla devam edilerek parolanın tamamı olan ‘h0pe’ elde edilerek sistem erişimi sağlanmıştır. Sisteme yapılan tüm girdiler ve işlem süreleri Tablo 4.5‘te verilmiştir.

Tablo 4.5: Sisteme yapılan parola tahmin girdileri ve girdilerin işlem süreleri.

Yapılan Girdi	Süre(µs)
\n	2,440
a	2,442
aa	2,440
aaa	2,444
aaaa	2,438
aaaaa	2,442
b	2,438
2	2,440
h	3,658
ha	3,662
hh	3,654
h1	3,656
h0	4,876
h0a	4,874
h00	4,880
h0p	6,096
h0pa	6,100
h0pp	6,096
h0pe	Sistem erişimi

Lojik analizör kullanılarak yapılan işlem sürelerinin ölçülerek analiz edilmesiyle parolanın elde edilmesi işleminin yanında parola hakkında bilgi edinmek için mikrodenetleyiciye ait parola doğrulama aşamasındaki harcanan güç osiloskop yardımıyla elde edilip güç sinyali incelenmiştir. Klavyeden basılan ‘enter’ tuşu sonrası mikrodenetleyicinin güç profili elde edilmiş ve yanlış karakterlerin benzer güç profili olduğu, parolaya ait karakterlerin güç profillerinin farklı olduğu görülmüştür. Bu şekilde mikrodenetleyicinin yapılan girdiler sonrası harcadığı güç analiz edilerek de sistem parolası elde edilmiştir.



Şekil 4.21: 'a' ve 'h' girdileri sonrası osiloskop ile elde edilen mikrodenetleyiciye ait güç profilleri

4.1.6. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parolanın Kelime Olarak Karşılaştırılması

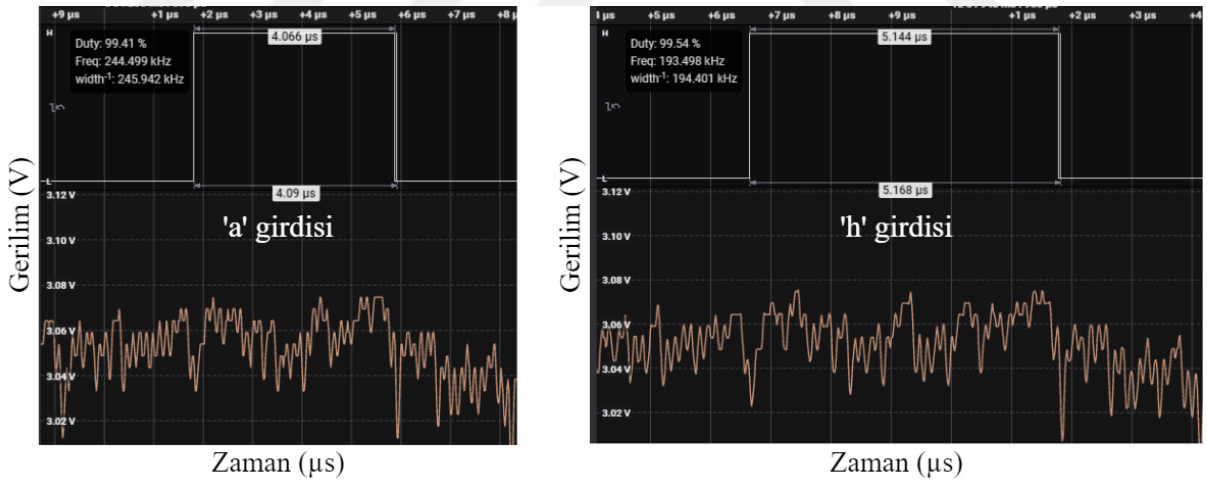
XMEGA mikrodenetleyici üzerinde parola kontrolünün kelime olarak karşılaştırılması yapılarak doğrulamanın sağlandığı sistem oluşturulup parolanın elde edilmesi işlemi için zamanlama analizi yapılmıştır.

```
if(strcmp(systemPassword, password)==0)
{
    my_puts("\nSystem access successful.\n");
    led_ok(1);
```

Şekil 4.22: Parola doğrulaması için yapılan girdi ve sistem parolasının kelime olarak karşılaştırılması.

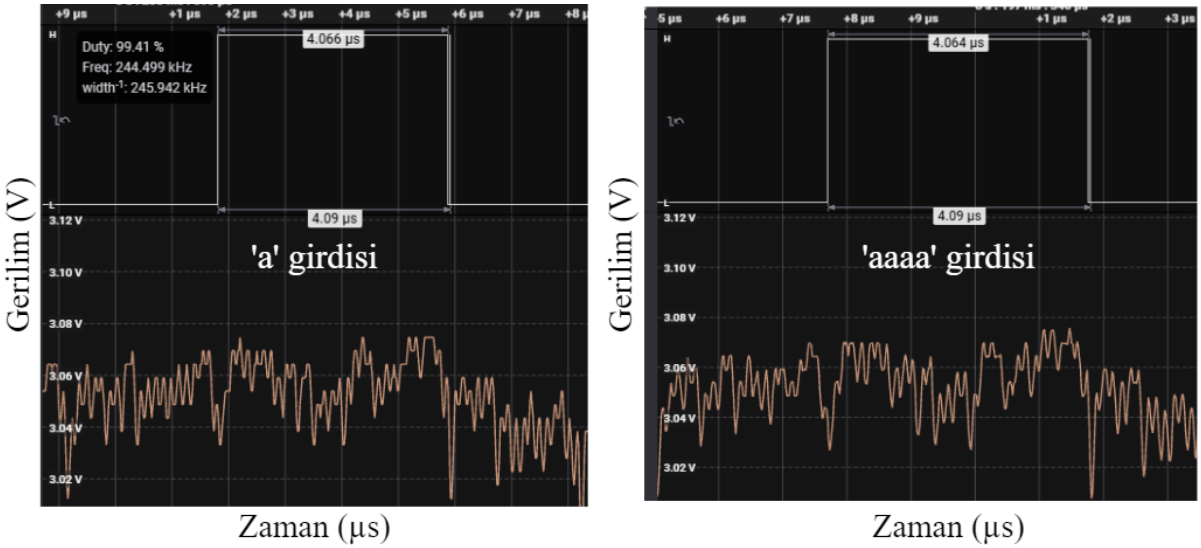
XMEGA mikrodenetleyicisinde parola korumalı sistem C dili ile yazılmıştır. Parola, kod içerisinde karakter dizisi (char array) olarak tanımlanmıştır. Kelime olarak karşılaştırma yapılması için ‘string’ veri tipinde karşılaştırma yapılmak istenmiş ve bu sebeple parola kontrolü ‘strcmp’ komutu kullanılarak yapılmıştır.

Sistemin parolasının elde edilmesi için ilk olarak harf ve rakamlar denenmiş ve ‘h’ harfinin işlem süresinin yapılan diğer girdilerin işlem süresinden daha uzun olduğu görülmüştür.



Şekil 4.23: 'a' ve 'h' girdilerinin lojik analizör ile ölçülen işlem süreleri.

Bölüm 4.1.2’de olduğu gibi kelime karşılaştırma yapısında parolanın kaç haneli olduğu bilgisi için farklı hanelerde girdiler uygulanmış ancak işlem süreleri arasında bir farklılık gözlemlenmemiştir.



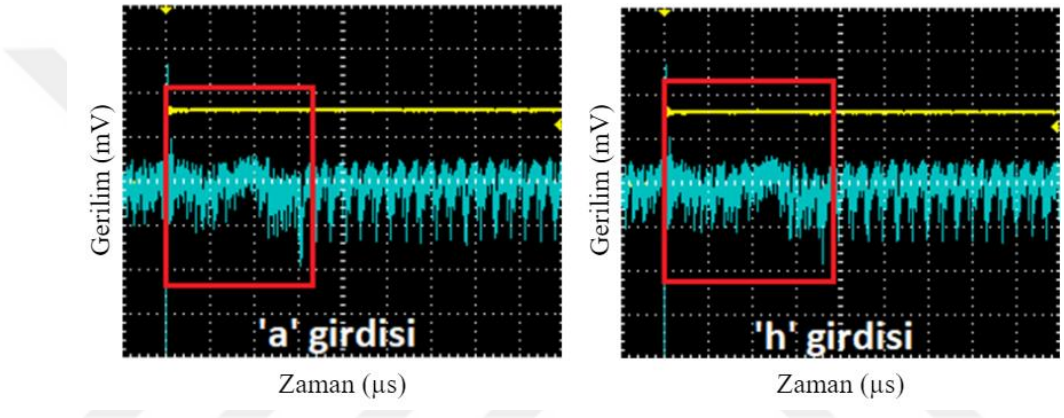
Şekil 4.24: 'a' ve 'aaaa' girdileri için lojik analizör ile ölçülen işlem süreleri.

İlk harf elde edildikten sonra ilk harf sabit tutulup ikinci harf denemeleri yapılarak (‘ha’, ‘hb’ gibi) işlem süreleri ölçülmüş ve bu şekilde her adımda ilgili adıma ait parola karakteri elde edilerek parolaya ulaşılmıştır. Tablo 4.6’da parolanın elde edilmesi için sisteme yapılan girdiler ve işlem süreleri verilmiştir.

Tablo 4.6: Parolayı elde etmek için sisteme yapılan girdiler ve işlem süreleri.

Yapılan girdi	Süre(µs)
\n	4,066
a	4,066
5	4,062
h	5,144
aa	4,064
aaa	4,062
aaaa	4,064
aaaaa	4,062
ha	5,146
hh	5,144
h1	5,154
h0	6,226
h0a	6,226
h00	6,238
h0p	7,310
h0pa	7,318
h0pp	7,312
h0pe	sistem erişimi

Klavyeden ‘enter’ tuşuna basılmasıyla uyarı mesajı çıkması arasındaki sürenin lojik analizör yardımıyla ölçülmesi işleminin yanında XMEGA mikrodenetleyicisinin harcadığı güç miktarı osiloskop yardımıyla analiz edilerek parola hakkında bilgilere ulaşılmıştır. Yapılan hatalı girdilere karşılık güç profili benzerlik gösterirken doğru karakter girildiği zaman mikrodenetleyicinin güç profilinde değişiklik olduğu görülmüştür. İşlem lojik analizör ile işlem süresi ölçümüne benzer şekilde her adımda elde edilen güç sinyali analiz edilerek bir karakter tespit edilmiş ve bu şekilde ilerleyerek sistem parolasına erişilmiştir. Şekil 4.25’te görüleceği üzere ‘a’ ve ‘h’ girdileri sonrasında mikrodenetleyicinin güç sinyalleri arasında belirgin bir fark ortaya çıkmaktadır.



Şekil 4.25: Mikrodenetleyiciye yapılan 'a' ve 'h' girdileri sonrası osiloskop ile elde edilen güç profilleri.

4.1.7. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parolanın Hash Değerinin Karşılaştırılması

Sistem üzerinde parolanın açık metin yerine daha güvenli bir yol olan hash değerinin tutulması senaryosu XMEGA mikrodenetleyici ile yapılan sistem üzerinde uygulanmıştır. Parolanın MD5 algoritması ile hash değeri elde edilmiş ve sistem içerisinde hash değeri tutularak parola doğrulama yapısı kullanıcının girdiği parolanın hash değeri ile sistem parolasının hash değerinin karşılaştırılması ile sağlanmıştır. Bu test senaryosunda hash değeri ile parola doğrulaması yapılan sistemin zamanlama analizi yapılarak parolası elde edilmeye çalışılmıştır.

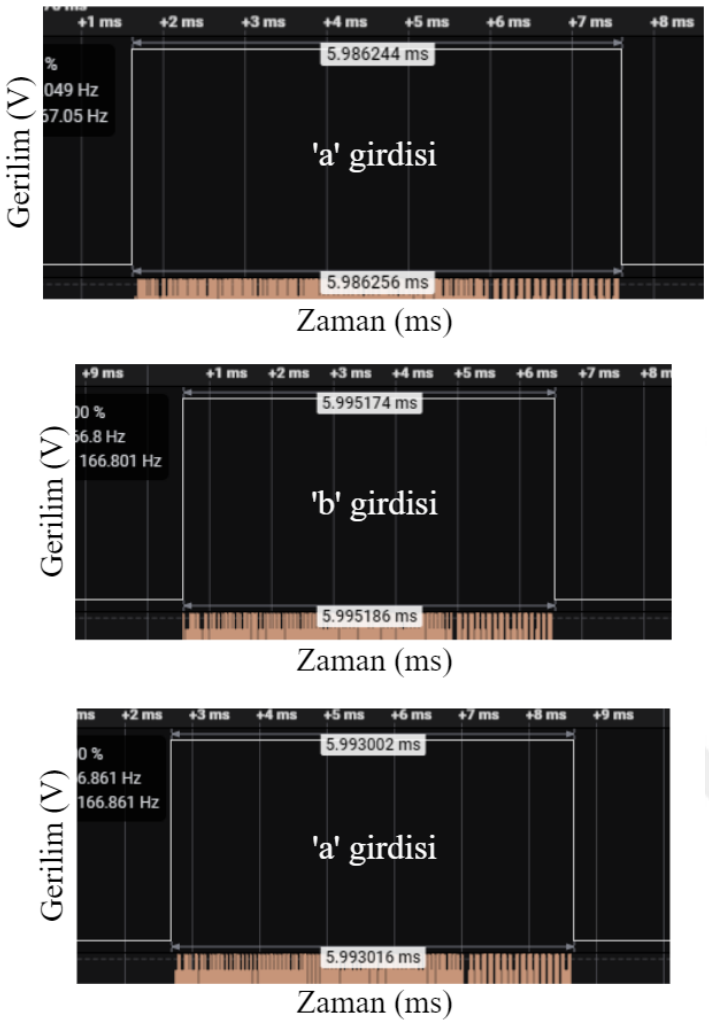
```
// hash işlemi
  uint8_t *passworddHash = md5String(passwd);

  for(unsigned int i = 0; i < 16; ++i){
      pos += sprintf(pos, "%02x", passworddHash[i]);
  }

  if(strcmp(sonuc,systemPassword_hash)==0)
  {
      my_puts("\nParola dogru.\nErisim saglandi!\n");
      led_ok(1);
```

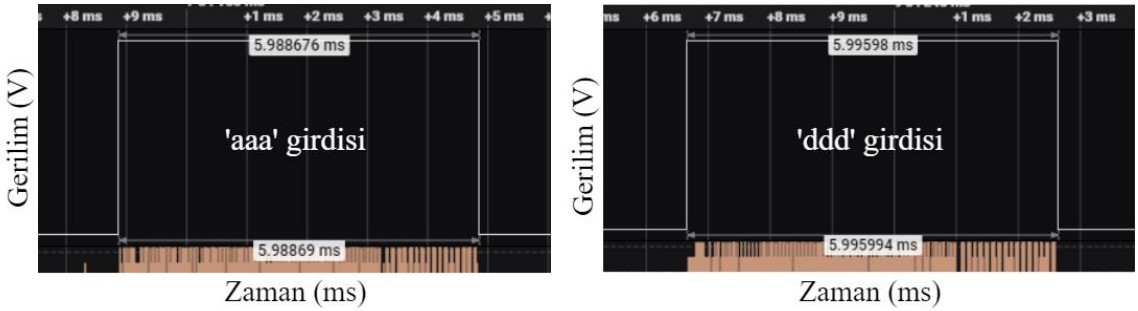
Şekil 4.26: Parolaların hash değerlerinin karşılaştırılarak parola doğrulaması yapılması.

İlk olarak XMEGA mikrodenetleyicisine harf ve rakam tahminleri gönderilerek işlem süreleri incelenmiş ve doğru harfi tespit edebilecek zaman farkı tespit edilememiştir.



Şekil 4.27: 'a', 'b' ve 'h' girdileri için lojik analizör ile elde edilen işlem süreleri.

Harf ve rakam olarak yapılan işlem süresi ölçümlerinde parolanın ilk karakterini elde edebilecek bir farklılık tespit edilememesinin ardından parolanın karakter sayısı hakkında bilgi edinmek için farklı uzunluklarda girdiler yapılarak işlem süreleri incelenmiş ancak diğer girdilerden farklı olabilecek bir sonuç elde edilememiştir. Aynı karakter uzunluğunda girdi yapıldığında dahi farklı işlem süreleri elde edilmiş ve karakter sayısı – işlem süresi arasında bir bağlantı tespit edilememiştir.



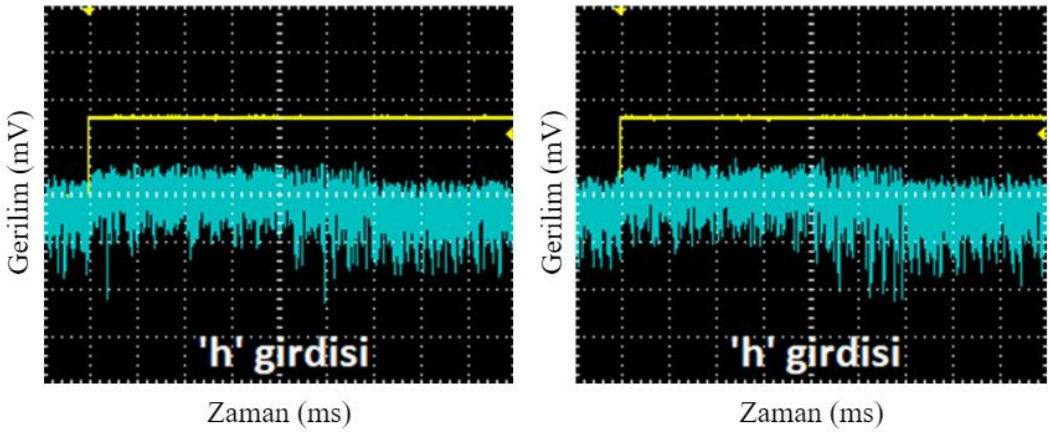
Şekil 4.28: Aynı karakter uzunluğuna sahip girdiler sonrasında elde edilen farklı işlem süreleri

Mikrodenetleyici üzerinde işlem sürelerinin elde edilip parola hakkında bilgi edinmek için yapılan girdiler ve işlem süreleri Tablo 4.7'de verilmiştir.

Tablo 4.7: Mikrodenetleyici üzerinde yapılan girdiler ve ölçülen işlem süreleri.

Yapılan Girdi	Süre(ms)
\n	5,971466
a	5,986244
aaa	5,988676
ddd	5,995980
aaaa	5,988004
eeee	5,981494
aaaaa	5,988678
b	5,995174
h	5,993002
cdf	5,981360
mxw	5,989224
prswqa	5,991792
ha	5,995306
h0	5,994490
h00	5,989212
h0p	5,995452
h0pa	5,981220

Lojik analizör ile yapılan 'enter' tuşuna basılmasıyla ekranda uyarı mesajının çıkması arasındaki sürenin ölçümü sonrasında elde edilen işlem sürelerinde belirgin bir farklılık tespit edilemediğinden bu test ile parola hakkında bilgi edinilememiştir. Mikrodenetleyicinin yapılan girdiler sonrasında yaptığı işlemler için harcadığı güç osiloskop yardımıyla incelendiğinde de aynı sonuç elde edilmiş ve girdilere ait güç sinyalleri arasında parolayı elde etmemizi sağlayacak bir farklılık tespit edilememiştir. Şekil 4.29 incelendiğinde aynı girdi sonrasında dahi farklı bir güç profili elde edilmiştir.

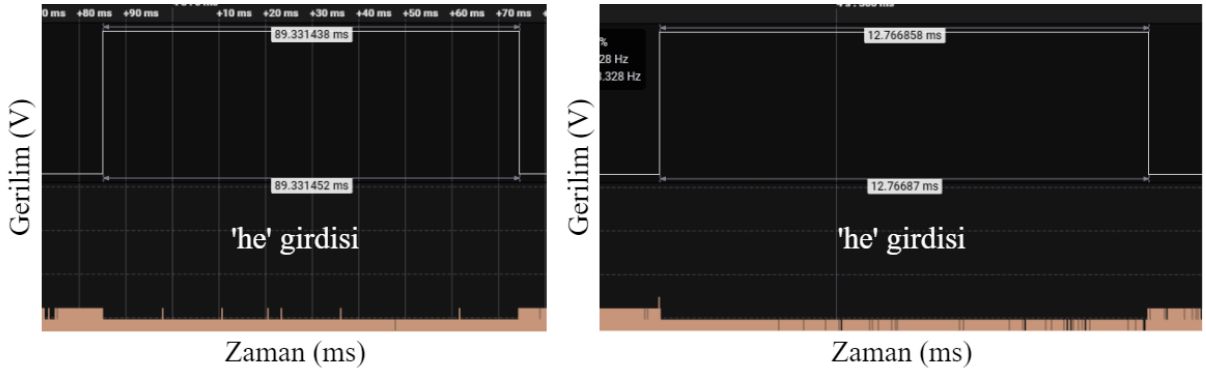


Şekil 4.29: 'h' girdisinin iki farklı zamanda denenmesiyle mikrodenetleyici üzerinden elde edilen güç profili.

4.1.8. XMEGA128D4-AU Mikrodenetleyici Üzerinde Parola Karşılaştırma Yapısına Rastgele Zaman Gecikmesi Koyularak Parolanın Kontrol Edilmesi

Sistem üzerinde çalışan yapıların işlem sürelerini rastgeleleştirmek ve işlemler hakkında dışarıya bilgi vermemek için kullanıcının girdiği parola ile sistem parolasının karşılaştırıldığı yapıya rastgele gecikme eklenmiş ve parolanın güvenliği değerlendirilmiştir.

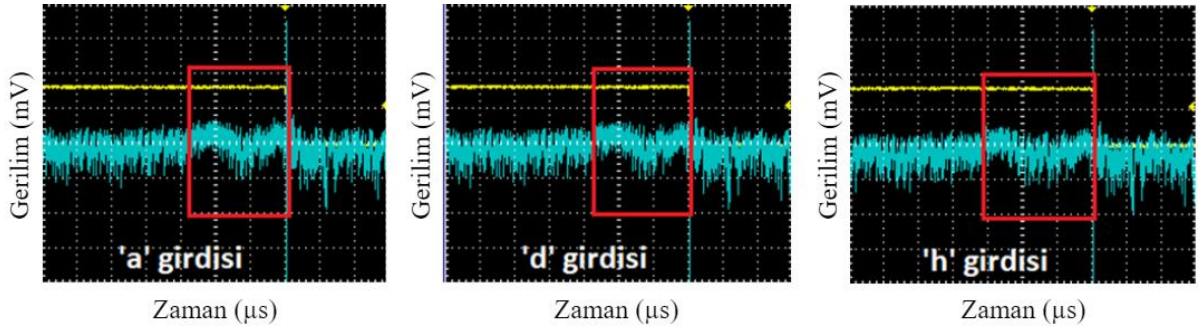
Yapılan girdilerin işlem süreleri klavyeden ‘enter’ tuşuna basılmasıyla ekranda uyarı mesajının çıkması arasındaki süre ölçülerek elde edilmiştir. Sistem içerisinde rastgele gecikme işlemi bulunduğundan dolayı aynı girdi tekrar yapıldığında dahi farklı işlem süresi elde edilmiş ve parola bu test ile elde edilememiştir.



Şekil 4.30: 'he' girdisinin farklı zamanlarda girilmesi sonrası elde edilen işlem süreleri.

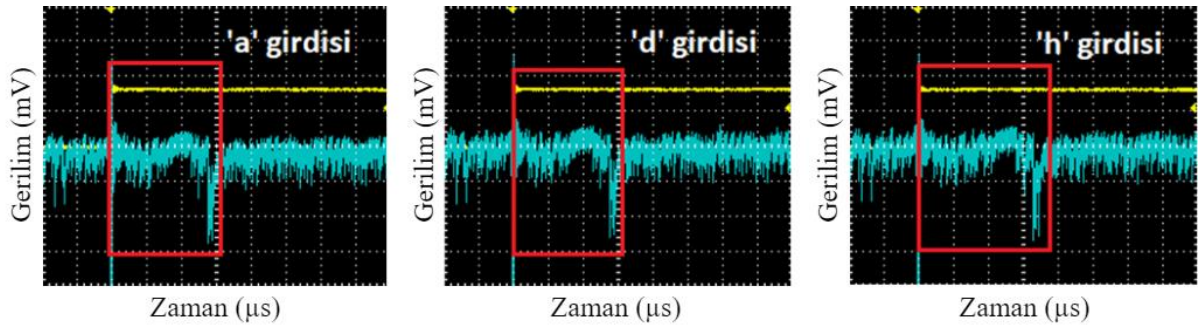
İşlem süresinin ölçülmesi işlemi kullanılarak rastgele zaman gecikmesi olan sistemde parola hakkında bilgi elde edilememiştir. Bu işlemin ardından mikrodenetleyicinin yapılan girdiler sonrası güç profili incelenerek parola elde edilmeye çalışılmıştır. Rastgele gecikmenin parola kontrolü yapısı öncesinde yapıldığı senaryoda, osiloskop yardımıyla yapılan güç ölçümleri

incelendiğinde ekrana uyarı mesajı çıkmadan hemen önceki durumda elde edilen güç sinyalinin parola hakkında bilgi verdiği ve parolaya ait olan karakterlerde farklı bir güç profili oluştuğu görülmüştür.



Şekil 4.31: 'a', 'd' ve 'h' girdileri sonrası ekrana uyarı mesajı çıkma anındaki mikrodenetleyiciye ait güç profili.

Parolanın ilk karakteri elde edildikten sonra 2 haneli parola denemeleri (‘ha’, ‘hb’ gibi) yapılarak parolanın ikinci hanesi ve bu işlem diğer haneler için tekrarlanarak parolanın tamamı güç sinyali incelenerek elde edilmiştir. Rastgele zaman gecikmesinin parola kontrolü sonrası ekrana uyarı mesajı göstermeden önce koyulması durumunda ise klavyeden ‘enter’ tuşuna basıldıktan sonra mikrodenetleyicinin harcadığı güç osiloskop yardımıyla elde edilmiş ve güç profili incelenmiştir. Girilen hatalı karakterlerin aynı veya benzer kalıpta güç profiline sahip olduğu parolaya ait ilk karakter girildiğinde ise farklı bir güç profili oluştuğu gözlenmiştir.



Şekil 4.32: 'a', 'd' ve 'h' girdileri yapıldıktan hemen sonra elde edilen mikrodenetleyiciye ait güç profili.

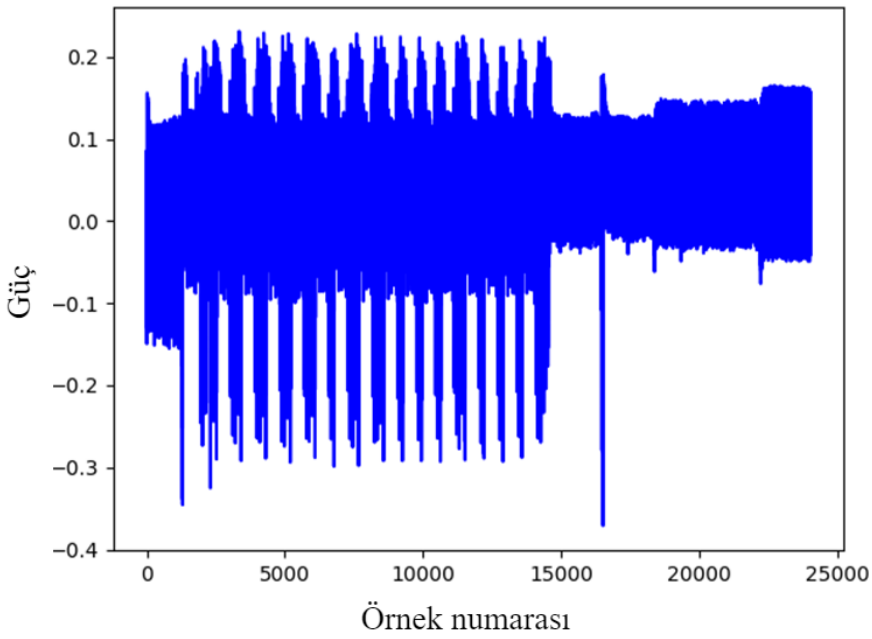
Parolanın ilk karakteri elde edildikten sonra ilk karakter sabit tutulup 2 haneli parola denemeleri (‘ha’, ‘hb’, ‘hc’ gibi) yapılarak parolanın ikinci hanesi ve parolanın diğer haneleri için aynı işlem devam ettirildiğinde güç profili incelenerek parolanın tamamı elde edilmiştir.

4.2. RSA ŞİFRELEME GERÇEKLEŞTİREN DONANIMDA GÜÇ ANALİZİ SALDIRISI

Chipwhisperer platformu kullanarak RSA şifreleme işlemi gerçekleştiren XMEGA128D4-AU mikrodenetleyici üzerinde güç analizi saldırısı gerçekleştirilmiştir. Mikrodenetleyicinin çalışması sırasında harcadığı güç elde edilip şifreleme işleminin gerçekleştiği aralıktaki sinyal analiz edilerek şifreleme anahtarı hakkında bilgi elde edilmeye çalışılmıştır. Çalışma için hazırlanan senaryoda mikrodenetleyici üzerinde 16-bit boyutunda anahtara sahip RSA şifreleme şifrelemesi çalıştırılmış ve gizli anahtarın kullanıldığı şifre çözme adımında sisteme tetikleme kodu eklenmiştir. Şifre çözme adımında tetikleme kodu XMEGA128D4-AU mikrodenetleyicisinin değerlendirilmesi için kullanılan CW308 UFO Kartı üzerinde GPIO4/TRIG pinini 'HIGH' (lojik 1) durumuna çekmiştir. İlgili pin 'HIGH' konumuna geçtiğinde mikrodenetleyicinin güç izi yakalanarak analiz yapılmıştır.

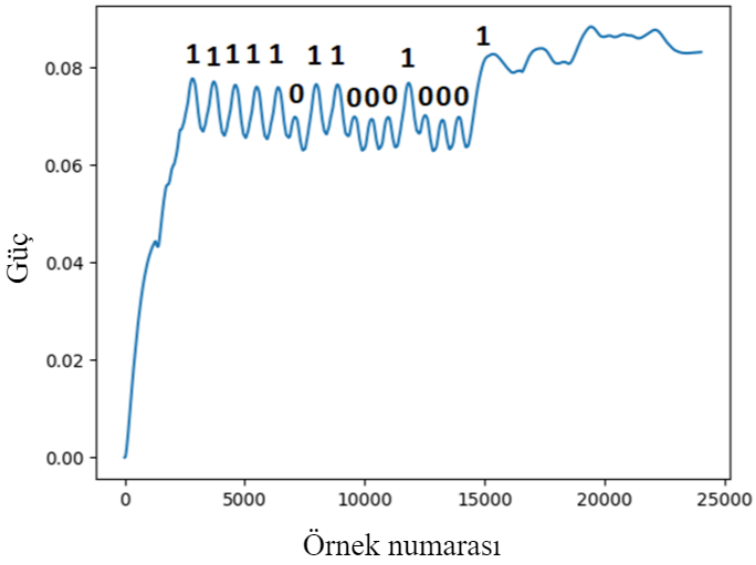
4.2.1. Basit Güç Analizi ile XMEGA128D4-AU Mikrodenetleyicisi Üzerinden Gizli Anahtarın Elde Edilmesi

RSA şifreleme işlemi yapan sisteme gizli anahtar değeri 16 bit boyutunda 'hexadecimal' formatta FB11 olarak verilmiş ve sistem çalıştırıldığında şifre çözme adımına ait sinyal Chipwhisperer donanımı ve Chipwhisperer Jupyter ortamı kullanılarak yakalanmıştır.



Şekil 4.33: XMEGA üzerinde RSA şifre çözme adımına ait güç izi.

Şekil 4.33'te verilen güç izi gözle incelendiğinde şifre çözme işleminin nerede yapıldığı sinyalin sadece bir bölümünde oluşan tepe ve dip noktalar dikkate alınarak görülebilmektedir. Ancak bu sinyale doğrudan bakarak anahtar hakkında bilgi elde edilememiştir. Bu güç izine alçak geçiren 'butterworth' filtresi uyguladığımızda Şekil 4.34'teki görülen sinyal elde edilmektedir.



Şekil 4.34: RSA şifre çözme adımına ait güç izinin alçak geçiren filtre uygulanmış hali.

Şekil 4.34'teki güç sinyali incelendiğinde sinyalde değişiklik gösteren her tepenin bize anahtar biti hakkında bilgi vereceği bilgisinden yola çıkarak her yüksek tepe 1 ve her alçak tepe 0 olarak alınmıştır. Bunun nedeni 1 durumunun mikrodenetleyici üzerinde daha fazla güç harcayacağı düşüncesidir. Bu şekilde sinyaldeki tepeler değerlendirilerek gizli anahtarın 'binary' formatta 1111101100010001 olduğu görülmüştür. Bu değer 'hexadecimal' olarak FB11 değerine karşılık gelmektedir.

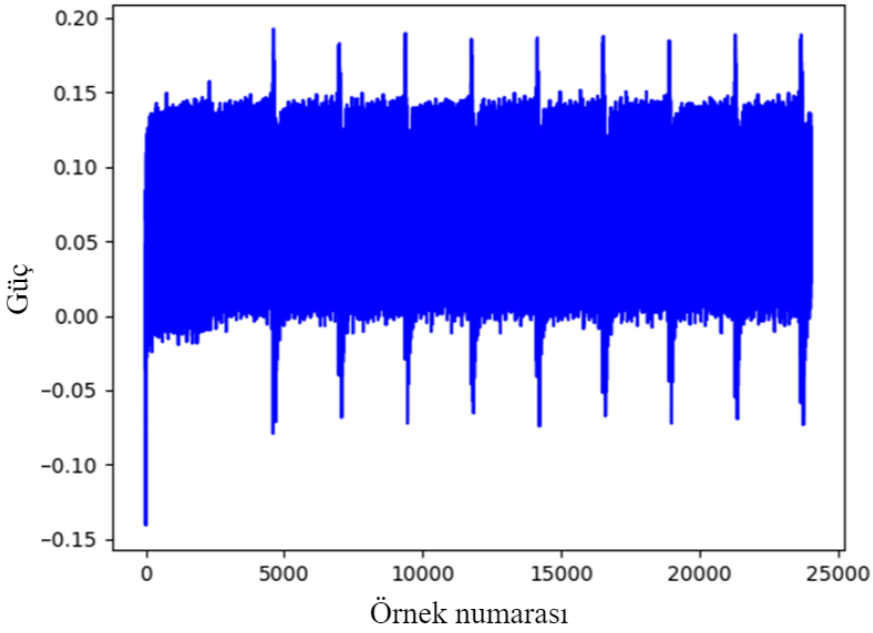
4.2.2. XMEGA128D4-AU Mikrodenetleyici Üzerinde RSA Şifreleme Adımlarına Anlamsız Rastgele Döngülerin Eklenmesi Durumunda Basit Güç Analizi

Mikrodenetleyici içerisindeki RSA şifreleme kodlarında şifreleme adımlarının içerisine mikrodenetleyicide güç tüketimine sebep olacak anlamsız işlemler eklenerek şifre çözme işlemine ait güç sinyalinin analizi yapılmıştır.

```
wait = j * rand() % 100;
for(volatile unsigned int ii = 0; ii < wait; ii++)
{
    float redundant = 2554*26987; //////////
}
```

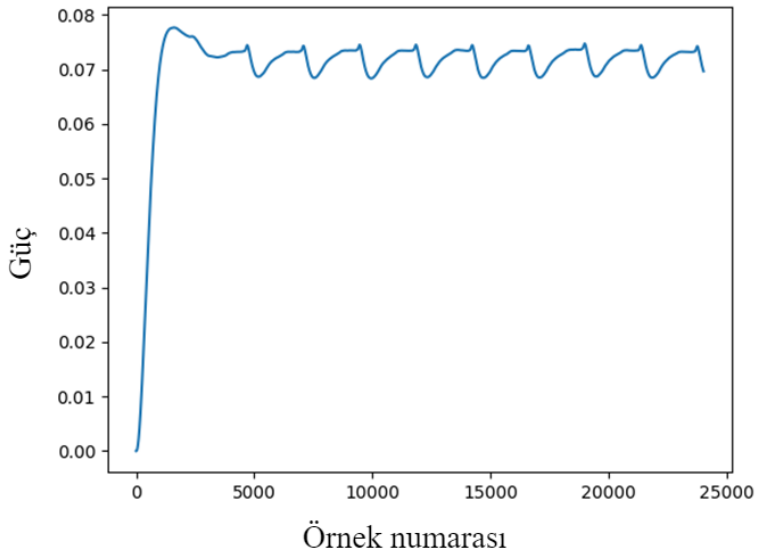
Şekil 4.35: Şifreleme algoritmasına ait kodların içerisine eklenebilecek anlamsız kod parçacığı örneği.

İlk olarak sisteme gizli anahtar değeri ‘hexadecimal’ olarak FB11 verilmiştir. Şifre çözme adımındaki güç izi yakalanmış ve incelenmiştir.



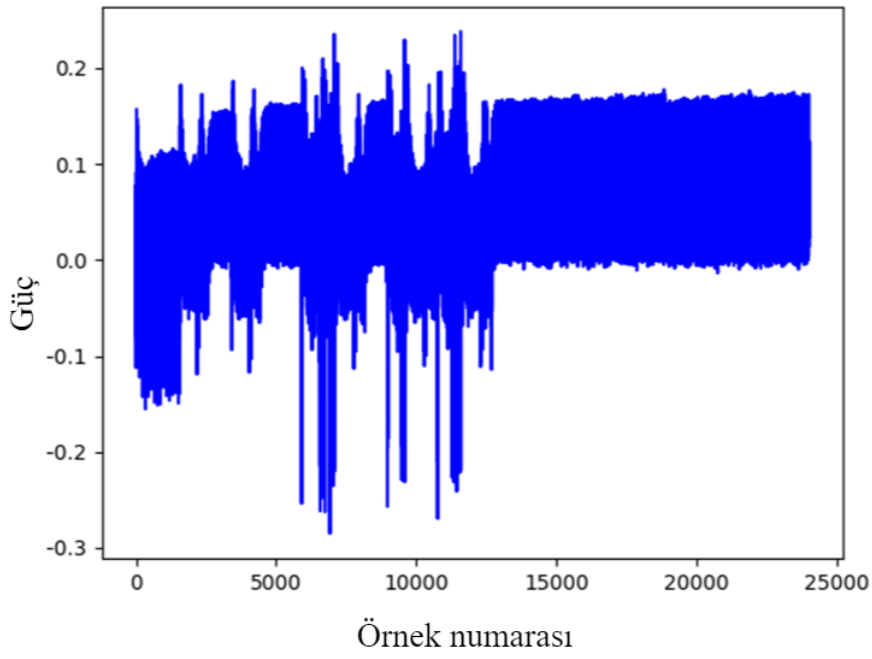
Şekil 4.36: Anlamsız kod parçacığının eklendiği sistemde şifre çözme adımı başladığı andan itibaren yakalanan güç izi.

Şekil 4.36’daki güç izi incelendiğinde harcanan güç miktarının belirli noktalarda tepe belirli noktalarda dip yaptığı görülmüştür. Ancak bu anahtar elde etmek için yeterli değildir. Güç izine Bölüm 4.2.1’de yapılan alçak geçiren ‘butterworth’ filtresi uygulandığında Şekil 4.37’de gösterilen sinyal elde edilmekte ve bu sinyal üzerinde anahtar bilgimizin yer almadığı görülmektedir.



Şekil 4.37: Şekil 4.36'da yer alan güç izinin alçak geçiren filtre uygulanmış hali.

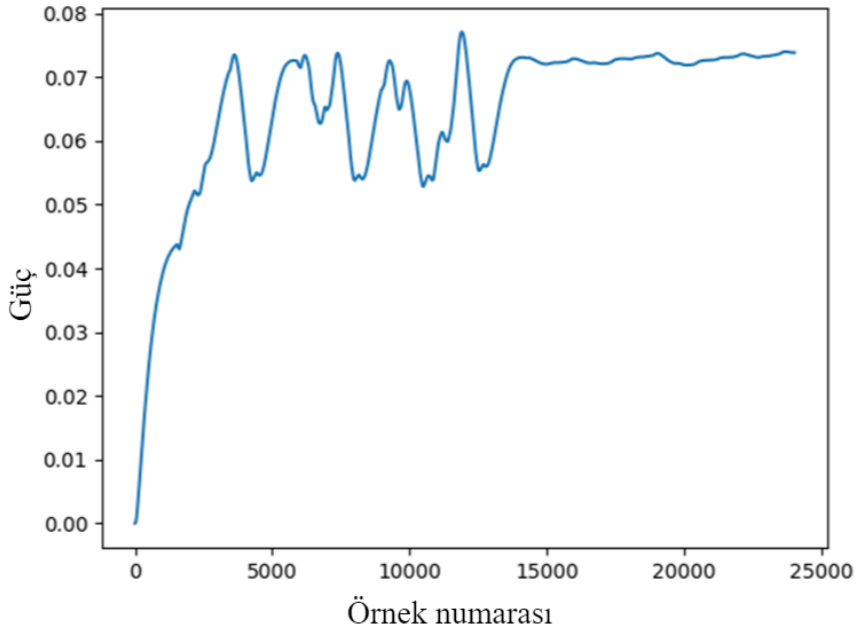
Aynı sistem üzerinde anlamsız kod parçacığının yeri ve döngü tur sayıları değiştirilerek (kod içerisinde şifreleme işlemlerine etki etmeyecek ama güç izinde şifreleme adımlarındaki güç izine etki edecek bir yere koyularak) yeniden çalıştırılmıştır. Anahtar değeri 'hexadecimal' formatta AA01 verilerek güç izi yakalanmıştır.



Şekil 4.38: Şifreleme algoritmasına anlamsız kod parçacığı eklenmiş sistemin güç izi.

Şekil 4.38 incelendiğinde şeklin başlangıcından orta bölümlere kadar güç izinde çok oynama olduğu sonrasında daha durağan ilerlediği görülmüştür. Oynamaların olduğu bölümde birçok

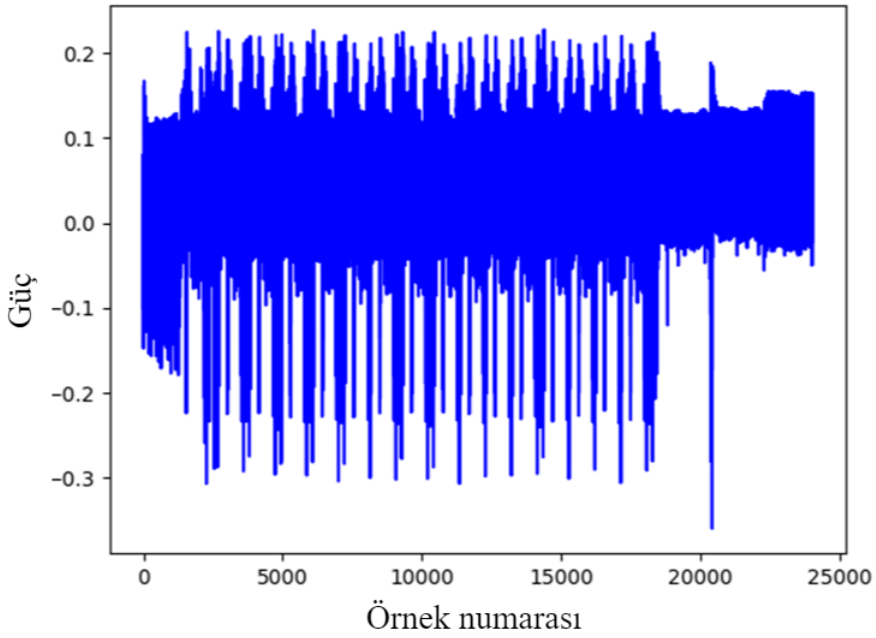
tepe nokta ve dip noktanın oluştuğu görülmüş ve burada bir işlem yapıldığı anlaşılmıştır. Sinyal içeriği hakkında daha temiz bilgiye sahip olmak için alçak geçiren ‘butterworth’ filtre uygulanmıştır.



Şekil 4.39: Şekil 4.38'de yer alan güç izinin alçak geçiren filtre uygulanmış hali.

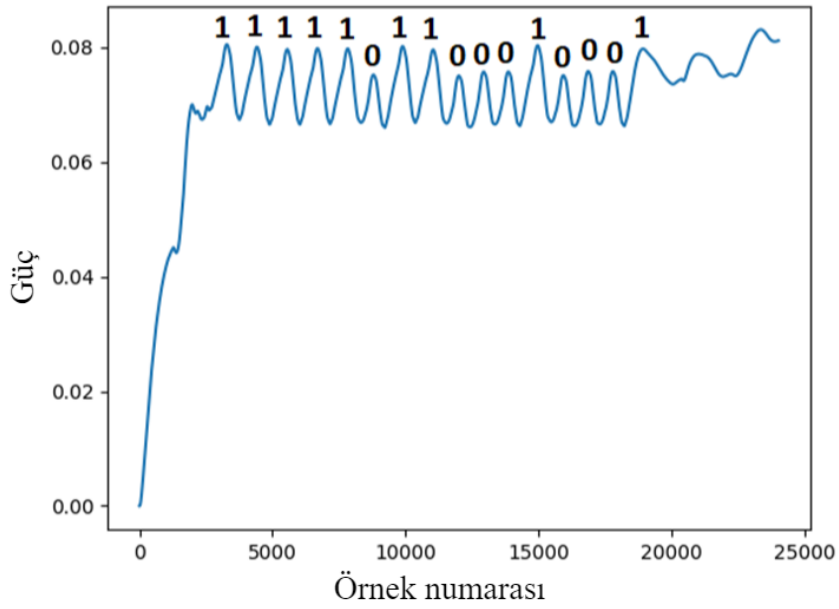
Şekil 4.39’da yer alan filtrelenmiş sinyal incelendiğinde yüksek ve alçak tepeler ile oluşturulan ‘binary’ değer atama işlemi sonrası elde edilen tahminin (yükselen tepelere 1, alçak tepelere 0 dediğimiz durumda sinyalin ‘binary’ – 1011010110011 ‘hexadecimal’ - 16B3 olduğu tahmin edilmiştir) sisteme girmiş olduğumuz anahtar ile bir bağlantısının olmadığı tespit edilmiştir.

Yapılan bir başka örnekte ise anlamsız kod parçacığındaki döngü tur sayısından rastgele sayı üretme yapısı kaldırılarak sabit uzunlukta düşük değerde tur sayısı verilmiş ve anahtar değeri ‘hexadecimal’ olarak FB11 olarak verilmiştir. Yakalanan sinyal incelendiğinde Şekil 4.33’teki sinyale benzer yapıda olduğu fakat daha uzun işlem yapıldığı tespit edilmiştir.



Şekil 4.40: Anlamsız kod parçacığındaki döngü tur sayısının azaltılmasıyla elde edilen güç izi.

Şekil 4.40'ta gösterilen güç izi alçak geçiren 'butterworth' filtreden geçirildiğinde ise elde edilen sinyalde şifreleme anahtarı hakkında bilgi olduğu tespit edilmiştir. Şekil 4.41'de elde edilen anahtar bilgisi sinyal üzerinde gösterilmiştir.



Şekil 4.41: Şekil 4.40'ta yer alan güç izinin alçak geçiren filtre uygulanmış hali.

5. TARTIŞMA

IoT cihazları internete bağlandıkları ve internet üzerinden veri alışverişi yapabildikleri için saldırganların hedefi haline gelmekte ve birçok farklı senaryoda saldırılara maruz kalabilmektedir. IoT cihazları için güvenliğin sağlanması konusunda cihazın donanımı da önemli bir yere sahiptir. Cihaz yazılımı donanım üzerinde çalıştığından kritik bilgiler donanım üzerinde tutulabilmektedir. Donanım içerisinde yer alan cihazın mikrodenetleyicisi yapılan işlemler veya işlediği veriler hakkında dışarıya bilgi sızdırabilmektedir. Saldırganlar mikrodenetleyiciden sızan bu bilgileri kullanarak kritik bilgileri ele geçirebilmektedir.

Erişim kontrolü gereken akıllı cihazlarda bulunan parola koruması ile izinsiz ve yetkisiz girişlerin önüne geçilebilir. Öte yandan ilgili cihazın donanımı çıkarılıp işlem süreleri incelendiğinde parola hakkında bilgi verebilir. Sisteme yapılan parola tahminlerinin mikrodenetleyici üzerinde işlenme ve sonrasında sonuç verme işlemleri arasındaki süre ölçülerek parola hakkında bilgi elde edilebilir. Buradaki düşünce parolaya ait doğru karakter dışındaki tüm karakterlerin benzer işlem süresine ve güç profiline sahip olacağıdır. Parolaya doğru karakter işlendikten sonra yazılım içerisindeki doğrulama yapısı bir sonraki karaktere geçerek karşılaştırma işlemine devam eder. Böylelikle parola tahminindeki doğru karakter sonrasında bir sonraki karaktere geçileceğinden işlem süresi daha uzun olabilir. Yanlış karakter karşılaştırıldığında yanlış olarak değerlendirilip hemen cevap dönecektir. Bu sebeple yanlış karakterler aynı veya benzer işlem süresine sahip olabilirler. Bunun yanında mikrodenetleyicinin parola doğrulaması sırasında harcadığı güç de aynı şekilde değişiklik gösterebilir. Ayrıca şifreleme işlemi gerçekleştiren mikrodenetleyicinin yine harcadığı güç incelenerek yaptığı işlemler hakkında bilgi edinilebilir. Şifreleme işlemi içerisinde belirgin güç harcayacak işlemler ve bu işlemlere karşılık gelen bitler belirlenerek elde edilen güç sinyali üzerinden şifreleme anahtarı elde edilebilir.

Mikrodenetleyicilerin parola koruması ve şifreleme işlemleri sırasındaki güvenlik durumlarını değerlendirmek için hazırlanan tez çalışmasında zamanlama analizi ve basit güç analizi saldırıları yapılmıştır. Zamanlama saldırısı için iki farklı mikrodenetleyici seçilmiş ve mikrodenetleyicilerin içerisindeki sistemde parola doğrulama mekanizması oluşturulmuştur. Hazırlanan yazılımlar ATmega328P mikrodenetleyicisi için Arduino diliyle,

ATXMEGA128D4-AU mikrodenetleyicisi için C dili ile yazılmıştır. Yazılım içerisinde farklı parola doğrulama yapılarının güvenlik durumlarını değerlendirmek için 4 farklı doğrulama senaryosu oluşturulmuştur. Mikrodenetleyicilerin kullanıldıkları değerlendirme kartları USB (Universal Serial Bus) ile bilgisayara bağlanmış ve aynı zamanda bilgisayardan aldığı güç ile çalışmıştır. Bilgisayardan açılan seri port üzerinden mikrodenetleyici kartlarına parola tahmini gönderilmiştir. Parola tahminlerinin işlem süreleri için klavyeden 'enter' tuşuna basıldıktan sonra ekranda uyarı mesajı çıkana kadar geçen süre ölçülmüş ve değerlendirilmiştir. Ölçüm lojik analizör ile yapılmış ve belirlenen iki noktanın ölçülmesi için mikrodenetleyici yazılımları içerisinde bu iki nokta için tetikleme sinyali oluşturulmuş ve geliştirme kartı üzerindeki bir pine çıkış verilmiştir. Lojik analizör ile bu çıkış ölçülerek işlem süreleri elde edilmiştir. İşlem süresinin yanında mikrodenetleyicinin topraklama pini ile geliştirme kartının topraklaması arasına direnç koyularak osiloskop ile güç izi yakalanmış ve bilgisayardan yapılan parola tahminleri sonrası 'enter' tuşuna basıldıktan sonra mikrodenetleyicinin harcadığı güç yakalanıp incelenerek parola elde edilmeye çalışılmıştır. Buradaki amaç hem zamanda gecikmeyi güç sinyallerinde görmek hem de doğru parola girildiğinde yanlış tahminlerden farklı bir güç profili oluşturduğu, yanlış tahminlerin birbirine benzer güç profili oluşturduğu fikrini doğrulamaktır. Basit güç analizi saldırısı için üzerinde RSA şifreleme algoritması çalışan bir mikrodenetleyici kullanılmıştır. Yapılan saldırı yan kanal saldırıları için hazırlanmış özel bir donanım kullanılarak gerçekleştirilmiş ve ayrıca mikrodenetleyici içerisindeki şifreleme işleminde güç analizinin zorlaşacağı düşünülen anlamsız ama güç harcayan kodlar sisteme eklenerek anahtar elde etmenin zorlaşma durumu da değerlendirilmiştir.

Yan kanal saldırıları donanım içerisinde gerçekleşen işlemler ve veriler hakkında bilgi sunarken uygulama aşamasında analiz işleminin yapılması kolay olmayabilir. Yan kanal saldırıları ile cihaz içerisindeki parolanın elde edilmesi veya şifreleme anahtarının elde edilmesi gibi işlemler için öncelikle cihazın donanımına fiziksel erişimin olması gerekir. Güç ölçümü için mikrodenetleyici geliştirme kartından sökülmeli ya da mikrodenetleyicinin Vcc veya GND bacağının kart ile bağlantısı sökülmeli ve araya direnç koyulmalıdır. Bacakları dışarıda olmayan BGA (Ball Grid Array – Bilyalı Izgara Dizisi) tip bir mikrodenetleyicide bu işlem çok daha zor hale gelebilir. Bu tip donanımlarda güç analizi yerine mikroişlemci veya mikrodenetleyici üzerinden yayılan bir elektromanyetik sinyal yakalanarak analiz yapılabilir. Mikrodenetleyicinin harcadığı güç sinyali osiloskop veya başka bir araçla elde edildiğinde karşılaşılan sinyalin hangi işleme ait olduğu mikrodenetleyicinin o an ne yaptığı bilgisi

anlaşılamayabilir. İstenilen işlemin güç profilinin mikrodenetleyicinin güç sinyali içerisinde bulunması yapılacak analiz için en önemli adımlardan birisidir. Bu sinyal elde edildikten sonra sisteme ait şifreleme işlemi hakkında bilgi edinilebilir. Sinyal analiz edilirken farklı çalışma ortamlarından veya güç kaynaklarından kaynaklı elde edilen sinyal gürültülü olabilir. Bu durumda analiz işlemi zorlaşabilir hatta sonuç vermeyebilir. Bu gürültünün gerek donanımsal filtre temizlenmesi ile gerekse de elde edilen güç izinin filtre yazılımından geçirilip temiz bir sinyal elde edilmesiyle analiz daha kolay hale gelebilir. Güç analizinde hangi noktadan itibaren sinyalin yakalanması gerektiği, zamanlama analizinde süre ölçümü için sistemde ölçülecek iki nokta arasının belirlenmesi sinyali analiz etmek ve sonuç çıkarmak için önemlidir. Ölçümün tam bu noktalarda alınması ile her girdi aynı şekilde değerlendirilebilir, girdiler ve parola arasında ilişki kurulabilir. Tez çalışmasında güç analizinde şifreleme işleminin başladığı nokta ve zamanlama analizinde ölçüm yapılacak iki nokta yazılım içerisinde işaretlenmiş, donanım üzerinde çıktı verilen pin lojik analizör ile dinlenerek değerlendirme yapılmıştır. Bu işlemle parola doğrulaması adımında rastgele zaman gecikmesi bulunan sistem değerlendirildiğinde işlem süreleri arasında, rastgele gecikmeler olduğundan, bağlantı kurulamayabilir. Bu durumda mikrodenetleyicinin güç profili incelenerek test yapılabilir. Bununla alakalı örnek senaryo tez çalışması kapsamında yapılmış ve parola elde edilmiştir. Tüm bu sistemler kullanıcıdan girdi kabul eden sistemler olduğundan girdi yapıldıktan sonra elde edilen sinyal ile işlem yapılarak değerlendirme yapılabilir. Tez çalışmasında hedef donanımların bilgisayar bağlantıları yapılmış ve girdiler seri port üzerinden hedef donanıma gönderilerek işlemler gerçekleştirilmiştir. Değerlendirme aşamasında belirlenen noktalardan birisi bu sebeple klavyeden ‘enter’ tuşuna basılmasıdır. Ayrıca testlerde elde edilen ölçümler donanımlar arasında değişiklik gösterebilir. Bu sebeple bir donanımda yapılan analiz işlemi bir başka donanım için kullanışlı olmayabilir. Bu durum mikrodenetleyicilerin farklı olmasından kaynaklandığı gibi mikrodenetleyici üzerinde çalışan yazılımın programlandığı dilden de kaynaklanabilir.

Yapılan deneysel testler gerek parola korumasında gerekse de şifreleme işleminde yazılan kodun donanım üzerindeki etkilerine odaklanmış ve mikrodenetleyici üzerinden alınan ölçümler ile güvenlik değerlendirmesi yapılmıştır. Bu işlem için senaryo gereği yazılım içerisinde ölçüm alınacak referans noktalar belirlenmiş ve böylece elde edilen ölçümler sadece istenilen işleme ait bilgileri vermiştir. Gerçek hayat uygulamalarında bu testler gerçekleştirilirken yazılım içerisine müdahale edilemeyeceğinden bu referans noktaların tanımlanması zor olabilir. Bir cihazın güç ölçümü yapıldığında elde edilen sinyalin hangi anda

hangi işlemi yaptığı bilinmediğinden bir saldırı yapmak ve saldırı sonucu doğru sonuçlara erişebilmek zor olabilir. Bu durumda parola tahmini veya şifreleme işlemine yan kanal saldırısı yapılmak istendiğinde kullanıcının yaptığı girdi anı başlangıç noktası olarak kabul edilerek ölçüm alınabilir. Yapılan girdi sonrası mikrodenetleyici işlem yaptığından dolayı o noktadan itibaren ölçülen işlem süresi veya elde edilen güç profili mikrodenetleyicide gerçekleşen işlemler hakkında bilgi verebilir. Öte yandan bu kullanıcıdan girdi alan bir sistem için geçerli bir işlem adımıdır. Kullanıcıdan girdi kabul etmeyen sistemde başlangıç referans noktası belirlenemeyebilir. Ayrıca kullanıcı girdisi sonrası mikrodenetleyici içerisinde çalışan yazılımda hemen şifreleme işlemi veya parola doğrulaması yapılmıyorsa öncesinde başka işlemler yapılıyorsa da belirlenen referans noktası sonrası elde ettiğimiz ölçümler doğru sonuca ulaştıramayabilir. Bu durumda daha uzun bir ölçüm yapıp birden fazla girdi için sinyal değişim oranlarına bakılarak analiz yapılabilir ve bilgi elde edilebilir. Hedef donanım üzerinde referans noktası birçok farklı şey olabilir. Örneğin parola korumalı kapı kilitlerinde yanlış girilen parolalar sonrası cihazın üzerinde kırmızı ışık yanıyorsa, parola tahmini girildikten ne kadar süre sonra kırmızı ışığın yandığı ölçülerek zamanlama analizi yapılabilir. Bu örnekte referans noktası cihaz üzerindeki uyarı ışığının aktif hale geçme süresidir.

Cihazlar üzerinde bu saldırıları önlemek için çeşitli önlemler alınabilir. Bunlar; bilgilerin hash değerlerinin tutulması, doğrulamanın hash üzerinden yapılması, doğrulama sistemlerinin sonuçlarının rastgele gecikme veya sabit gecikme sonrası gösterilmesi, sistem içerisine anlamsız ama güç tüketen kod parçacıklarının eklenmesi, yazılım içerisinde dallanmaların artırılması, donanımın kurcalamaya karşı dayanaklı hale getirilmesi gibi önlemler olabilir. Yapılan tez çalışmasında parolanın hash olarak doğrulamasının yapılması deneyinde parola bilgisi elde edilememiştir.

Hazırlanan tez çalışmasında yapılan zamanlama analizi testleri parola koruması yapan ve yapılan girdiler sonucu uyarı veren bir sistemde uygulanabilir. Cihazın verdiği uyarı referans noktası alınıp süre ölçümü yapılarak parola hakkında bilgi edinilebilir. Aynı zamanda mikrodenetleyicinin güç profili girdi yapıldıktan hemen sonra yakalanarak analiz edilebilir ve bu şekilde yine parola hakkında bilgi edinilebilir. Elde edilen sonuçlara göre hangi senaryo ile saldırı yapılacağı belirlenerek sistem içerisinden bilgi edinme süresi azaltılabilir. Örneğin test yapılan bir cihazda girilen karakterlerin işlem süreleri arasında bir farklılık oluşmuyorsa yapılan girdinin karakter sayısı her denemede bir artırılarak önce parolanın kaç haneli olduğu bulunabilir ve sonrasında parolaya ulaşılabilir. Ya da yapılan ölçümlerde aynı girdi

yapıldığında dahi farklı işlem süresi ölçülüyorsa bu durumda mikrodenetleyicinin harcadığı güç ölçülerek parola elde edilebilir. Basit güç analizi konusunda yapılan deney ise RSA şifreleme işlemi gerçekleştiren bir donanım üzerinde uygulanabilir. Elde edilen güç sinyali filtreden geçirilerek anlamlandırılabilecek sinyal elde edilebilir ve bu sinyal üzerinden her yüksek tepe noktalara 1, alçak tepe noktalara 0 diyerek anahtar elde edilebilir. Yapılan deneysel çalışmanın amaçlarından birisi olan basit güç analizini engelleme yöntemi olarak gerçek hayattaki donanımlara ait yazılım içerisinde anlamsız işlemler yaptırılarak saldırganın elde ettiği güç sinyali daha karmaşık hale getirilebilir. Bu durum basit güç analizini bir noktaya kadar engelleyebileceği gibi kesin koruma da sağlamayabilir.

6. SONUÇ VE ÖNERİLER

Donanım güvenliği kapsamında literatürde yapılan çalışmalarda birçok yan kanal saldırı türü denenmiş ve donanım içerisindeki gizli anahtar elde edilmeye çalışılmıştır. Hazırlanan tez çalışmasında literatürde yapılan çalışmalara ek olarak geliştirilen yazılımın hem parola koruması hem de şifreleme işleminin güvenliği konusunda donanım üzerindeki etkilerinin analiz edilmesi ve güvenliğinin hazırlanan yazılım kapsamında değerlendirilmesi amaçlanmıştır. Bu çalışmayla birlikte güvenli bir sistem gerçekleştirmek için temel güvenlik önlemleri yapılan deneylerle ortaya koyulmuştur. Zamanlama analizine karşı önlem olarak literatürde geçen koruma önlemleri yazılım üzerinde gerçekleştirilmiş ve hem işlem süresi hem de mikrodenetleyicinin işlem esnasındaki güç profilleri incelenerek güvenlik sağlama durumları değerlendirilmiştir. Bu işlemlerin hem farklı donanımlar üzerinde hem de farklı programlama dilleri üzerindeki etkileri ortaya koyulmuştur. Aynı zamanda güç analizi saldırılarına karşı önlem olarak sunulan yöntemin doğrulaması yapılarak yapılan saldırılar karşısında güvenlik durumu ortaya koyulmuştur. Gerçekleştirilen çalışmayla birlikte literatüre, donanım güvenliği kapsamında örnek IoT donanımındaki güvenlik durumları ve karşı önlemlerin farklı test senaryoları ile sistem üzerinde ne kadar güvenlik sağladığı konusunda katkı sunulması amaçlanmıştır.

Yapılan tez çalışmasında donanım güvenliği saldırı çeşitlerinden birisi olan yan kanal saldırıları seçilen mikrodenetleyiciler üzerinde denenmiştir. Güvenlik durumlarının değerlendirilmesi aşamasında deneysel testler için birçok farklı senaryo oluşturulmuş ve saldırılar düzenlenmiştir. Saldırı olarak yan kanal saldırıları arasından hem zamanlama analizi saldırısı hem de basit güç analizi saldırısı uygulamaları denenmiştir. Zamanlama analizi ile parola korumalı bir sisteme saldırı yapılmış ve sistem içerisinde bulunan parola farklı senaryolar üzerinden elde edilmeye çalışılmıştır. Basit güç analizi ile şifreleme işlemi gerçekleştiren donanıma saldırı yapılmış ve içerisinde bulunan şifreleme anahtarının elde edilmesi farklı senaryolar üzerinden değerlendirilmiştir.

Zamanlama analizi yaparak parolanın elde edilmesi için hazırlanan senaryolar iki farklı mikrodenetleyici üzerinde uygulanmıştır. Arduino programlama dili ile programlanan ATmega328P mikrodenetleyicisi ve C programlama dili programlanan ATXMEGA128D4-AU

mikrodenetleyicisi üzerinde çalıştırılan 4 senaryodan sadece hash ile doğrulama yapılan senaryoda parola güvenliğinin sağlandığı görülmüştür. Diğer 3 senaryoda sistem parolası elde edilmiş ve sistem erişimi sağlanmıştır.

Test edilen iki senaryoda açık metin halde parola doğrulama yapısı farklı karşılaştırma işlemleri ile yapılarak zamanlama analizleri değerlendirilmiştir. Yapılan değerlendirmede farklı karşılaştırma işlemlerinin mikrodenetleyici üzerinde farklı etkilerinin olduğu görülmüştür. Her iki mikrodenetleyicide karakter bazlı parola doğrulama mekanizmasın zamanlama analizinde karakter karakter parola elde edilirken ATmega328P üzerinde çalışan kelime bazlı parola doğrulama mekanizmasının zamanlama analizinde öncelikle parolanın karakter sayısı tespit edilmekte ardından parolanın karakterleri tespit edilmektedir. ATXMEGA128D4-AU mikrodenetleyicisinde üzerinde çalışan kelime bazlı doğrula mekanizmasının zamanlama analizinde parolanın karakter bazlı karşılaştırma senaryosunda olduğu gibi karakter karakter elde edildiği görülmüştür. ATXMEGA128D4-AU mikrodenetleyicisinde kelime bazlı karşılaştırma yapısı C dilinde bulunan ‘strcmp’ komutu ile yapılmıştır. Elde edilen sonuç incelendiğinde ‘strcmp’ komutunun karşılaştırma işleminin donanım üzerinde karakter bazlı karşılaştırma ile aynı etkiye sebep olduğu sonucu çıkarılmıştır. Her iki mikrodenetleyici içerisinden parolaların elde edilmesi sistemin güvensiz olduğunu göstermektedir.

Atmega328P mikrodenetleyicisindeki 4 haneli bir parolayı elde etmek için bulgularda yer alan gecikme süreleri göz önüne alınarak yapılan tüm girdilerin işlem süreleri (sistem 40 farklı karakter desteklediği durumda) karakter bazlı karşılaştırma senaryosunda yaklaşık 1.5 milisaniye, kelime bazlı karşılaştırma sistemi denendiğinde ise yaklaşık 1.4 milisaniye olarak hesaplanmıştır. ATXMEGA128D4-AU mikrodenetleyicisinde ise 4 haneli parola için yapılan tüm girdilerin süreleri (sistem 40 farklı karakter desteklediği durumda) karakter bazlı karşılaştırma senaryosunda yaklaşık 680 mikrosaniye olarak hesaplanmıştır. ATXMEGA128D4-AU mikrodenetleyicisindeki kelime bazlı karşılaştırma sisteminin ise analizi karakter bazlı sistemle aynı olduğu tespit edildiğinden işlem süresi aynı olmuştur. C programlama dili ile programlanan ATXMEGA128D4-AU mikrodenetleyicisinin karşılaştırma adımında daha hızlı işlem süresi olduğu sonucu da buradan çıkarılabilir.

Hash doğrulaması yapılan sistemde girilen parola tahminlerinin karakter sayısı fark etmeksizin aynı boyutta çıktı vermesi karakter sayısı bazlı karşılaştırma yapılmasını engellemektedir. Aynı zamanda bir girdinin hash alma işlemi belirli matematiksel işlemler ile yapılması ile

mikrodenetleyici üzerinde farklı güç profillerine sebep olmakta ve hash olmadan yapılan doğrulamaya göre daha uzun süre sürmektedir. Hash alma işlemi sonucu girilen karakterlerin değişmesiyle değişiklik göstermesi ve yapılan girdinin tek bir harfinin değişmesiyle dahi tamamen farklı bir çıktı üretmesi harf tahmini yapmanın önüne geçmektedir. Elde edilen sonuçlar değerlendirildiğinde parola koruması olan sistemlerde parolanın hash değerinin tutulması ve hash değeriyle doğrulama yapılması zamanlama analizi saldırılarına karşı sistemi güvenli hale getirmektedir.

Zamanlama analizi saldırılarına alınabilecek önlemlerden bir diğeri olan doğrulama işlemi öncesi veya sonrası rastgele zaman gecikmesi eklenmesi senaryosu her iki mikrodenetleyici üzerinde uygulandığında işlem sürelerinin analizi konusunda güvenlik sağlamaktadır. Yapılan parola tahminleri sonrası elde edilen işlem süreleri rastgele olması parolanın karakterlerini ve parolanın kendisini yapılan girdilerle tahmin etmeyi engellemektedir. Ancak bu durumun sadece işlem süresi analizinde bir güvenlik sağladığı görülmüş ve mikrodenetleyiciye yapılan parola tahmini girdileri sonrası mikrodenetleyicinin harcadığı güç yakalanarak analiz edildiğinde parola karakterlerinin elde edilebildiği tespit edilmiştir. Bu durum rastgele gecikme olsa dahi önce parola doğrulamasının yapılıp ardından gecikmenin işlenmesinden dolayı (veya önce rastgele gecikmenin işlenmesi ve ardından parola doğrulamasının yapılması) girilen yanlış karakterlerin benzer güç profili ile başlayacağı (ya da biteceği), doğru karakter girildiğinde farklı bir güç profili oluşacağı düşüncesinden kaynaklanmaktadır. Rastgele gecikme işlemi her girdi için aynı güç tüketimine sebep olacağından sinyalin bir bölümü benzerlik gösterecek ve parolanın elde edilebilmesi için analiz işlemini kolaylaştıracaktır. Bu sonuç göz önüne alındığında sistem üzerinde güvenlik sağlamak için doğrulama adımı öncesine veya sonrasına rastgele zaman gecikmesi koymak tek başına yeterli olmamaktadır. Buradaki bilgi sızıntısının doğrulama mekanizmasının kendisinde yer aldığı görülmektedir. Buradan hareketle alınacak güvenlik önleminin bu noktayı kavraması gerektiği söylenebilir.

Yapılan testlerde girdilerin işlem süreleri ölçülerek parola hakkında bilgi elde edilebileceğinin yanında mikrodenetleyicinin harcadığı güç incelenerek de parola hakkında bilgi toplanabileceği görülmüştür. Harcanan gücün analiz edilerek sistem içerisinden bilgi toplanan bir başka saldırı olan basit güç analizi saldırısı ile şifreleme yapan donanım üzerinden şifreleme anahtarı elde edilmiştir. Burada elde edilen bulgulardan birisi yakalanan sinyale doğrudan bakıldığında anahtar bilgisi tespit edilemediği ancak güç izinde şifrelemenin belirgin olduğudur. Sinyal alçak geçiren filtreden geçirilerek temiz bir sinyal elde edilmiş ve 1 bitlerinin daha fazla güç harcadığı

düşüncesinden yola çıkarak anahtar elde edilmiştir. Bu çalışmada amaçlanan bu sinyali yazılım içerisinde yaptığımız değişikliklerle daha karmaşık hale getirmek ve analizini zorlaştırmaktadır. Bunun için şifreleme işlemini etkilemeyecek anlamsız bir kod parçacığı yazılarak şifreleme adımlarının farklı yerlerine koyulmuş ve şifreleme adımının güç sinyali yakalanarak analiz edilmiştir. Elde edilen sinyallerde rastgele döngün tur sayılarına sahip kod parçacığı sonrasında elde edilen sinyalde anahtar hakkında bilgi elde etmenin zorlaştığı görülmüştür. Bu işlemin basit güç analizine karşı tam korum sağlamadığı son işlemle anlaşılmıştır. Yapılan son senaryoda anlamsız kod parçacığının döngü tur sayısı sabit ve düşük değerli tutulmuş ve şifreleme adımının güç sinyali yakalanmıştır. Elde edilen sinyal daha karmaşık durmasına karşın alçak geçiren filtreden geçirildiğinde şifreleme anahtarının tamamı elde edilmiştir. Bu sebeple yapılan karmaşıklaştırma işlemi tam güvenlik sağlamamakta, sadece belirli örneklerde analizi zorlaştırmaktadır.

KAYNAKLAR

[1]. Mattern, F., Floerkemeier, C., 2010, From the Internet of Computers to the Internet of Things, *From Active Data Management to Event-Based Systems and More*, Lecture Notes in Computer Science, vol 6462, In: Sachs, K., Petrov, I., Guerrero, P. (ed.), Springer, Berlin, Heidelberg, 242-259.

[2]. Forbes, 2002, *The internet of things*, https://www.forbes.com/global/2002/0318/092.html?sh=753904e23c3e, [Ziyaret Tarihi: 15 Eylül 2022].

[3]. Ashton, K., 2009, That 'internet of things' thing, *RFID journal*, 22(7), 97-114.

[4]. INFSO D.4 networked enterprise and RFID INFSO G.2 Micro & Nanosystems, Co-operation with the working group RFID of the ETP EPOSS, 2008, Internet of Things in 2020: A Roadmap For The Future, Berlin: Information Society and Media, Tech. Rep.

[5]. Perera, C., Liu, C.H., Jayawardena, S., 2015, The Emerging Internet of Things Marketplace From an Industrial Perspective: A Survey, *IEEE Transactions on Emerging Topics in Computing*, 3(4), 585- 598.

[6]. Alaba, F.A., Othman, M., Hashem, I.A.T., Alotaibi, F., 2017, Internet of Things security: A survey, *Journal of Network and Computer Applications*, 88, 10-28.

[7]. Evan, D., 2011, *The Internet of Things How the Next Evolution of the Internet Is Changing Everything*, https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf, [Ziyaret Tarihi: 7 Ekim 2022]

[8]. Lopez, J., Rios, R., Bao, F., Wang, G., 2017, Evolving privacy: From sensors to the internet of things, *Future Generation Computer Systems*, 75, 46-57.

[9]. Stojkoska, BLR, & Trivodaliev, KV., 2017, A review of internet of things for smart home: Challenges and solutions, *Journal of Cleaner Production*, 140, 1454–1464.

[10]. HaddadPajouh, H., Dehghantanha, A., Parizi, R.M., Aledhari, M., & Karimipour, H., 2021, A survey on internet of things security: Requirements, challenges, and solutions, *Internet of Things*, 14, 100-129.

[11]. Song, Y., Yu, F.R., Zhou, L., Yang, X., & He, Z., 2021, Applications of the Internet of Things (IoT) in Smart Logistics: A Comprehensive Survey, *IEEE Internet of Things Journal*, 8(6), 4250-4274.

[12]. Vailshery, L.S., 2022, *Internet of Things (IoT) total annual revenue worldwide from 2019 to 2030*, https://www.statista.com/statistics/1194709/iot-revenue-worldwide/, [Ziyaret Tarihi: 18 Eylül 2022].

[13]. Vailshery, L.S., 2022, *Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2021, with forecasts from 2022 to 2030*, https://www.statista.com/statistics/1183457/iot-connecteddevices-worldwide/, [Ziyaret Tarihi: 18 Eylül 2022].

[14]. Carey, C., 2022, *EV charging points hacked to show explicit material*, https://cities-today.com/ev-charging-points-hacked-to-show-explicit-material/, [Ziyaret Tarihi: 3 Ağustos 2022]

[15]. Kelion, L., 2018, *MiSafes' child-tracking smartwatches are 'easy to hack'*, https://www.bbc.com/news/technology-46195189, [Ziyaret Tarihi: 3 Ağustos 2022]

[16]. Kelion, L., 2018, *Swann's home security camera recordings could be hijacked*, https://www.bbc.com/news/technology-44809152, [Ziyaret Tarihi: 3 Ağustos 2022]

[17]. Carey, C., 2022, *Major UK bus operator hit by cyber attack*, https://cities-today.com/major-uk-bus-operator-hit-by-cyber-attack/, [Ziyaret Tarihi: 5 Ekim 2022]

[18]. Onag, G., 2022, *Forescout reveals 56 flaws in OT devices from 10 companies*, https://futureiot.tech/forescout-reveals-56-flaws-in-ot-devices-from-10-companies/, [Ziyaret Tarihi: 5 Ağustos 2022]

[19]. https://www.forescout.com/resources/ot-icefall-report/, [Ziyaret Tarihi: 7 Ekim 2022]

[20]. Sava, J.A., 2022, *Global number of IoT cyber attacks 2020-2021*, https://www.statista.com/statistics/1322216/worldwide-internet-of-things-attacks/, [Ziyaret Tarihi: 12 Ekim 2022]

[21]. Grau, A., 2015, Can you trust your fridge?, *IEEE Spectrum*, 52(3), 50-56.

[22]. Rubin, S., 2016, *When coffee makers attack*, https://www.computerworld.com/article/3135874/when-coffee-makers-attack.html, [Ziyaret Tarihi: 12 Ekim 2022]

[23]. Bertino, E., Islam, N., 2017, Botnets and Internet of Things Security, *Computer*, 50(2),76-79.

[24]. https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf, [Ziyaret Tarihi: 16 Ağustos 2022]

[25]. https://wiki.owasp.org/index.php/OWASP_Internet_of_Things_Project#tab=IoT_Vulnerabilities, [Ziyaret Tarihi: 16 Ağustos 2022]

[26]. Koley, S., Ghosal, P., 2015, Addressing Hardware Security Challenges in Internet of Things: Recent Trends and Possible Solutions, *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*, 10-14 August 2015 Pekin, ISBN: 978-1-4673-7211-4, 517-520.

[27]. Azrour, M., Mabrouki, J., Guezzaz, A., Kanwal, A., 2021, Internet of Things Security: Challenges and Key Issues, *Security and Communication Networks*, 2021, 1-11.

[28]. Sadhu, P.K., Yanambaka, V.P., Abdelgawad, A., 2022, Internet of Things: Security and Solutions Survey, *Sensors*, 22(19), 7433.

[29]. Bhunia, S., Tehranipoor, M., 2018, *Hardware security: a hands-on learning approach*, Morgan Kaufmann, ISBN: 978-0-12-812477-2.

[30]. Skorobogatov, S., 2014, *Tamper resistance and hardware security*, https://www.cl.cam.ac.uk/~sps32/PartII_030214.pdf, [Ziyaret Tarihi: 16 Ağustos 2022]

[31]. Hu, W., Chang, C.-H., Sengupta, A., Bhunia, S., Kastner, R., Li, H., 2021, An Overview of Hardware Security and Trust: Threats, Countermeasures, and Design Tools, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 40(6), 1010-1038.

[32]. https://wiki.owasp.org/index.php/OWASP_Internet_of_Things_Project#tab=IoT_Attack_Surface_Areas, [Ziyaret Tarihi: 20 Ağustos 2022]

[33]. Vasile, S., Oswald, D., Chothia, T., 2019, Breaking All the Things—A Systematic Survey of Firmware Extraction Techniques for IoT Devices, *Smart Card Research and Advanced Applications*, 12–14 November 2018 Montpellier, Springer Cham, ISBN:978-3-030-15462-2, 171–185.

[34]. https://www.exploitee.rs/, [Ziyaret Tarihi: 20 Ekim 2022]

[35]. Sidhu, S., Mohd, B.J., Hayajneh, T., 2019, Hardware Security in IoT Devices with Emphasis on Hardware Trojans, *Journal of Sensor and Actuator Networks*, 8(3), 42.

[36]. Kalamkar, G., Gotkhindikar, A., Suryawanshi, A.R., 2018, Low-Level Memory Attacks on Automotive Embedded Systems, *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 16-18 August 2018 Pune, IEEE, 1-5.

[37]. Kumar, M., 2019, *Researchers Discover TPM-Fail Vulnerabilities Affecting Billions of Devices*, https://thehackernews.com/2019/11/tpm-encryption-keys-hacking.html, [Ziyaret Tarihi: 12 Ekim 2022]

[38]. Lakshmanan, R., 2022, *MIT Researchers Discover New Flaw in Apple M1 CPUs That Can't Be Patched*, https://thehackernews.com/2022/06/mit-researchers-discover-new-flaw-in.html, [Ziyaret Tarihi: 12 Ekim 2022]

[39]. National Cyber Security Centre, 2018, *Meltdown' and 'Spectre' guidance*, https://www.ncsc.gov.uk/guidance/meltdown-and-spectre-guidance, [Ziyaret Tarihi: 20 Ekim 2022]

[40]. Kessler, G.C., 2022, *An Overview of Cryptography*, http://www.garykessler.net/library/crypto.html, [Ziyaret Tarihi: 20 Ekim 2022]

[41]. Singh, G., Supriya, 2013, A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security, *International Journal of Computer Applications (0975 – 8887)*, 67(19), 33-38.

[42]. Coron, J.-S., 2006, What is cryptography?, *IEEE Security & Privacy*, 4(1), 70-73.

[43]. Jain, N., 2020, *Breaking DES using Differential Cryptanalysis*, https://medium.com/lotus-fruit/breaking-des-using-differential-cryptanalysis-958e8118ff41, [Ziyaret Tarihi: 20 Ekim 2022]

[44]. Kocher, P., Jaffe, J., Jun, B., 1999, Differential power analysis, *Advances in Cryptology - CRYPTO '99*, 15-19 August 1999 Santa Barbara, Springer, Berlin, Heidelberg, 388–397.

[45]. Spreitzer, R., Moonsamy, V., Korak, T., Mangard, S., 2018, Systematic Classification of Side-Channel Attacks: A Case Study for Mobile Devices, *IEEE Communications Surveys & Tutorials*, 20(1), 465-488.

[46]. Genkin, D., Shamir, A., Tromer, E., 2014, RSA key extraction via low-bandwidth acoustic cryptanalysis, *Advances in Cryptology -- CRYPTO 2014*, 17-21 August 2014 Santa Barbara, Springer, Berlin, Heidelberg, 444–461.

[47]. Gamaarachchi, H., Ganegoda, H., 2018, Power Analysis Based Side Channel Attack, arXiv preprint arXiv:1801.00932.

[48]. Kocher, P., Jaffe, J., Jun, B., Rohatgi, P., 2011, Introduction to differential power analysis, *Journal of Cryptographic Engineering*, 1, 5–27.

[49]. Bayam, K.A., ÖRS, B., 2010, Differential power analysis resistant hardware implementation of the RSA cryptosystem, *Turkish Journal of Electrical Engineering and Computer Sciences*, 18(1), 129-140.

[50]. Bayat-Makou, P., Jahanian, A., Reshadi, M., 2018, Security Improvement of FPGA Design Against Timing Side Channel Attack Using Dynamic Delay Management, *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, 13-16 May 2018 Quebec, IEEE, 1-4.

[51]. Kocher, P.C., 1996, Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems, *Advances in Cryptology — CRYPTO '96*, 18–22 August 1996 Santa Barbara, Springer, Berlin, Heidelberg, 104–113.

[52]. Karimi, E., Jiang, Z.H., Fei, Y., Kaeli, D., 2018, A Timing Side-Channel Attack on a Mobile GPU, *2018 IEEE 36th International Conference on Computer Design (ICCD)*, 07-10 October 2018 Orlando, IEEE, 67-74.

[53]. Chen, G., Zhu, Z., Mao, B., Chen, W., Jian, J., Zhang, Z., Zhuang, Y., 2020, A review of hardware timing channel detection and mitigation, *Journal of Physics: Conference Series*, 1634(1), 012118.

[54]. Brumley, D., Boneh, D., 2005, Remote timing attacks are practical, *Computer Networks*, 48(5), 701-716.

[55]. https://chipwhisperer.readthedocs.io/en/latest/getting-started.html, [Ziyaret Tarihi: 25 Ekim 2022]

[56]. https://store.arduino.cc/products/arduino-uno-rev3, [Ziyaret Tarihi, 26 Ekim 2022]

[57]. https://www.microchip.com/en-us/product/ATmega328P#, [Ziyaret Tarihi, 26 Ekim 2022]

[58]. https://www.microchip.com/en-us/product/ATxmega128D4, [Ziyaret Tarihi: 20 Kasım 2022]

[59]. https://github.com/newaetech/chipwhisperer-target-cw308t/tree/main/CW308T_XMEGA, [Ziyaret Tarihi: 20 Kasım 2022]

İNTİHAL RAPORU İLK SAYFASI

Mustafa_Sirin

ORİJİNALLİK RAPORU

%5	%4	%0	%3
BENZERLİK ENDEKSİ	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCİ ÖDEVLERİ

BİRİNCİL KAYNAKLAR

1	Submitted to The Scientific & Technological Research Council of Turkey (TUBITAK) Öğrenci Ödevi	%2
2	acikbilim.yok.gov.tr İnternet Kaynağı	%1
3	Submitted to Marmara University Öğrenci Ödevi	<%1
4	docplayer.biz.tr İnternet Kaynağı	<%1
5	acikerisim.nigde.edu.tr:8080 İnternet Kaynağı	<%1
6	web.itu.edu.tr İnternet Kaynağı	<%1
7	it-forensik.fiw.hs-wismar.de İnternet Kaynağı	<%1
8	Submitted to American University of the Middle East Öğrenci Ödevi	<%1
9	acikerisim.nevsehir.edu.tr	

KURUM İZNİ YAZILARI

Uyarı: Canlı ve cansız deneklerle yapılan tüm çalışmalar için kurum izin belgelerinin eklenmesi zorunludur. Gizlilik ve mahremiyet içeren durumlarda kurum adı kapatılmalıdır.

☐ **Kurum izni gerekmektedir.**

☒ **Kurum izni gerekmemektedir.**

Mustafa ŞİRİN
(İmza)