T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AKILLI EV CİHAZLARININ HABERLEŞMESİNDE HAFİF SIKLET ŞİFRELEME ALGORİTMALARININ PERFORMANS ANALİZİ

YÜKSEK LİSANS

Ömer YEL

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

ŞUBAT 2024

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AKILLI EV CİHAZLARININ HABERLEŞMESİNDE HAFİF SIKLET ŞİFRELEME ALGORİTMALARININ PERFORMANS ANALİZİ

YÜKSEK LİSANS

Ömer YEL

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Bilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Hüseyin ESKİ

ŞUBAT 2024

Ömer Yel tarafından hazırlanan "Akıllı Ev Cihazlarının Haberleşmesinde Hafif Sıklet Şifreleme Algoritmalarının Performans Analizi" adlı tez çalışması 20.02.2024 tarihinde aşağıdaki jüri tarafından oy birliği ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Bilim Dalı'nda Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı:

Jüri Üyesi:

Jüri Üyesi:

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "Akıllı Ev Cihazlarının Haberleşmesinde Hafif Sıklet Şifreleme Algoritmalarının Performans Analizi" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazetede yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(……/……/20…..).

(imza)

Öğrencinin Adı Soyadı

İÇİNDEKİLER

Sayfa

KISALTMALAR

AES : Advanced Encryption Standard

DES : Data Encryption Standard

NSA : National Security Agency

NIST : National Institute of Standards and Technology

IOT : Internet Of Things

ECB : Electronic Code Book

OFB : Output Feedback

CFB : Cipher Feedback

CBC : Cipher Block Chaining

CTR : Counter

TABLO LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

AKILLI EV CİHAZLARININ HABERLEŞMESİNDE HAFİF SİKLET ŞİFRELEME ALGORİTMALARININ PERFORMANS ANALİZİ

ÖZET

Akıllı ev sistemlerinde kesintisiz veri alışverişi, verimli ve güvenli bir ortamın omurgasıdır. Aydınlatma ve sıcaklığın kontrol edilmesinden güvenlik kameralarının izlenmesine kadar bu sistemlerin hepsi veri üreten ve paylaşan çok sayıda birbirine bağlı cihaza dayanır. Bu verilerin hem hızlı bir şekilde iletilmesini hem de güvenli bir şekilde korunmasını sağlamak çok önemlidir.

Hızlı ve enerji tasarruflu veri şifreleme ve iletişim, akıllı ev teknolojisinin iki temel özelliğidir. Bu sistemler genellikle enerjinin kısıtlı olduğu ortamlarda çalışır. Bu nedenle veri iletişim süreci minimum güç tüketecek, cihazların ömrünü uzatacak ve işletme maliyetlerini azaltacak şekilde optimize edilmelidir.

Veri şifreleme, akıllı ev kullanıcılarının mahremiyetini ve güvenliğini sağlamada çok önemli bir rol oynar. Yetkisiz erişimi önler ve güvenlik kamerası görüntüleri veya çeşitli sensörler tarafından toplanan kişisel veriler gibi hassas bilgileri korur. Ancak, gerçek zamanlı yanıtların ve sorunsuz çalışmanın sağlanması için bu şifreleme işleminin hızlı olması gerekir. Yavaş şifreleme, gecikmelere yol açarak akıllı ev ekosisteminin kusursuz işleyişini bozabilir.

Bu bağlamda hafif şifreleme algoritmalarının kullanılması önemlidir. Bu algoritmalar hem hız hem de enerji tüketimi açısından verimli olurken aynı zamanda güçlü bir güvenlik sağlayacak şekilde tasarlanmıştır. Hafif şifreleme yöntemlerini benimseyen akıllı ev cihazları, aşırı güç tüketimine veya önemli bir gecikmeye neden olmadan güvenli bir şekilde iletişim kurabilir.

Hızlı ve enerji açısından verimli veri iletişimi, akıllı ev cihazlarının gerçek zamanlı izlenmesi ve kontrolü için özellikle kritik öneme sahiptir. Örneğin, akıllı bir termostat ani bir sıcaklık değişimi tespit ederse, uygun ayarlamaların tetiklenmesi için bu bilgiyi hızlı bir şekilde merkezi kontrol ünitesine iletmelidir. Gecikmiş iletişim rahatsızlığa ve enerji verimsizliğine yol açabilir.

Akıllı ev sistemlerinde hızlı ve enerji tasarruflu veri şifreleme ve iletişim ihtiyacı göz ardı edilemez. Bu sistemlerin güvenilirliğini, güvenliğini ve verimliliğini korumanın anahtarı budur. Hafif şifreleme yöntemlerinden yararlanarak ve veri aktarımını optimize ederek, akıllı evlerin enerji kaynaklarını korurken ve işletme maliyetlerini azaltırken hayatlarımızı daha konforlu ve güvenli hale getirmeye devam etmesini sağlayabiliriz.

Bu çalışmada akıllı ev sistemlerinde veri iletişiminde kullanılan hafif şifreleme algoritmalarının performans karşılaştırması yapılmıştır. Akıllı ev sistemlerinde kullanılan cihazların simülasyonu için Nodemcu ve Raspberry Pi3 cihazlarından yararlanılmaktadır. Nodemcu cihazı, ev ortamında eylem gerçekleştiren ve veri üreten cihazları temsil ederken, Raspberry Pi cihazı ise merkezi kontrol ünitesini temsil ediyor. Simülasyon sisteminde akıllı ev cihazı, çalışması sırasında ürettiği verileri

merkezi kontrol merkezine gönderir. Veri aktarımı sırasında veriler hafif şifreleme algoritmaları kullanılarak şifrelenir. Veri şifreleme sürecinde hafif şifreleme algoritmaları olarak Ascon, Chacha, Speck ve Aes algoritmaları kullanılmaktadır. Şifreleme algoritmalarının performans ölçümleri gerçekleştirilmiştir. Blok şifreleme algoritmalarında (Aes ve Speck) 128bit, 192bit ve 256 bit anahtar uzunluklarında ve ECB, OFB, CFB, CTR ve CBC modlarında ayrı ayrı performans ölçümleri gerçekleştirilmiştir. Akış şifreleme algoritmalarında (Ascon ve Chacha) 128bit, 192bit ve 256bit anahtar uzunlukları ayrı ayrı uygulanarak performans ölçümleri gerçekleştirilmiştir. 512 byte'lık veri şifreleme ve şifre çözme işlem işlemi gerçekleştirilmiş ve bir byte'lık verinin işlem süresi üzerinden karşılaştırma yapılmıştır. Çalışma, hafif sıklet algoritmalarında anahtar uzunluklarının ve blok modlarının etkisini ve geleneksel şifreleme algoritmalarından olan Aes algoritmasıyla performans farklarının belirlenmesini amaçlamaktadır.

IN COMMUNICATION OF SMART HOME DEVICES PERFORMANCE ANALYSIS OF LIGHTWEIGHT ENCRYPTION ALGORITHMS

SUMMARY

Today, smart home systems and Internet of Things (IoT) devices stand out as technologies that radically change our lifestyle and make significant contributions to a number of usage areas. Smart home systems offer various benefits such as improving quality of life, optimizing energy efficiency and increasing security through home automation and connected devices. Homeowners can remotely control their heating and cooling systems through smart thermostats, save energy with smart lighting systems, and monitor their homes through security cameras.

IoT devices in smart home systems play an important role in data communication, and this process forms the basis of home automation. These devices collect various data through various sensors and connected hardware. For example, smart thermostats detect indoor and outdoor temperature data of the home, security cameras record images, and motion sensors monitor activities in the home.

Data communication between these devices usually occurs over wireless networks. The use of communication protocols such as Wi-Fi, Bluetooth, Zigbee and Z-Wave ensures fast and reliable data transfer between devices. This gives homeowners remote access via mobile apps or other smart home control tools. Thanks to this, users can monitor the status of their home, control devices remotely and even optimize their energy consumption.

Preferring Wi-Fi for data transmission between IoT devices in smart home systems is of great importance in providing a fast, reliable and wide-coverage communication infrastructure. Wi-Fi stands out as a wireless network technology and contributes to the interactive and smooth operation of smart home devices with its many advantages.

Data communication also enables interaction between devices in smart home systems. For example, a smart thermostat receives outdoor temperature data and automatically adjusts the home's interior temperature based on that information. Additionally, when security cameras detect movement, this information is reported to other devices, allowing them to react to a specific event, such as turning on lights or activating an alarm.

In smart home systems, using the MQTT (Message Queuing Telemetry Transport) protocol in communication between IoT devices plays an important role in ensuring effective and efficient data transfer. The reasons why MQTT is preferred are its lightweight design, broadcast/subscriber model, reliable communication, and flexible usage features.

First of all, MQTT's lightweight nature is highly advantageous for energy-limited IoT devices. Since these devices are generally battery-based or optimized to minimize

energy consumption, MQTT's low bandwidth requirement and low resource consumption contribute to longer battery life of these devices.

Additionally, MQTT's broadcast/subscribe model enables flexible and instantaneous data sharing between devices. The data that a device broadcasts on a topic can be instantly received by other devices, enabling real-time interaction between devices in the smart home. For example, when a motion sensor detects activity, this information can be quickly transmitted to other devices via MQTT so that other devices in the home can react accordingly.

However, security measures also play a critical role in this data communication process. Encrypting data and using secure wireless networks helps homeowners protect their personal information and the security of their homes. In this way, smart home systems offer their users both a more comfortable life and a safe technology experience.

In smart home systems, data carried between IoT devices plays a key role in ensuring the integration and functionality of the systems. This data enables users to manage their homes more efficiently, increase their security and optimize their overall living comfort.

Encrypted data transmission between IoT devices in smart home systems is of critical importance to protect users' privacy and home security. These systems enable various devices in the home to communicate with each other, providing a more comfortable and smart lifestyle. However, if encryption is not used during data transmission between these devices, sensitive data such as personal information and in-home activities can potentially fall into the hands of malicious individuals.

The use of lightweight encryption algorithms in data transmission between IoT devices in smart home systems provides significant advantages. These lightweight algorithms focus on low energy consumption while providing strong security. Since smart home devices often operate in energy-constrained environments, it is a critical factor that the encryption algorithms used in data transmission are lightweight and energy efficient.

There are various reasons why traditional encryption algorithms are not preferred in data transmission between IoT devices in smart home systems. These reasons include the fact that traditional encryption algorithms generally require excessive resource consumption and incompatibility with fast data transmission. Traditional encryption algorithms generally require high computing power and memory capacity, which is unsuitable for energy-constrained IoT devices. In addition, traditional encryption algorithms generally require high computing power and memory capacity. The large key lengths and complexity of encryption algorithms contrast with the limited processing capabilities of smart home devices. Low resource consumption is important because these devices are generally battery-based or optimized for energy efficiency. Since traditional encryption algorithms cannot work efficiently on such devices, lighter and energy-friendly encryption methods are preferred.

Lightweight encryption algorithms contribute to providing real-time responses by accelerating data transmission processes. Particularly in smart home applications, instantaneous data exchange between devices is often required. For example, a motion sensor must immediately transmit any activity it detects to the central control unit. Lightweight encryption algorithms enable such fast and continuous data transmission, minimizing delays and contributing to the smooth operation of smart home systems in daily use.

Additionally, energy-saving lightweight encryption algorithms help extend the battery life of smart home devices. Because these algorithms provide security by consuming less energy, so devices can be used for a longer time. This is especially important for battery-based devices or devices operating in energy-constrained environments.

The key length of encryption algorithms is a critical factor in the security of data. The key length determines the mathematical complexity and security level of the encryption algorithm. Typically, algorithms use 128-bit, 192-bit and 256-bit key lengths, providing different levels of security.

Algorithms with a 128-bit key length provide security that is generally considered sufficient for most applications today. 192-bit key length represents a more complex password structure than 128-bit and provides a higher level of security. The 256-bit key length represents the strongest level of encryption so far. Keys of this length offer a level of security that is nearly impossible to break with current technology.

ECB (Electronic Codebook), OFB (Output Feedback), CFB (Cipher Feedback), CTR (Counter), and CBC (Cipher Block Chaining) modes enable encryption algorithms to be adapted to the scenarios in which they are used.

In this study, the performance comparison of lightweight encryption algorithms used in data communication in smart home systems was made. The devices used in smart home systems were simulated. Nodemcu and Raspberry Pi3 devices were used in the simulation. While the Nodemcu device represents IoT devices that perform actions and produce data in the home environment, the Raspberry Pi device represents the central control unit. In the simulation system, the NodeMCU device uses the operating time, device settings and encryption algorithm selected via the user interface. operated according to the options. The Nodemcu device sent the data it produced during its operation to the central control device over the wireless network. MQTT broker is run on the Raspberry device. The Raspberry device monitors the issue at the broker as a central control device. It broadcasted from the Nodemcu device to the relevant topic on the MQTT broker. The central control unit decrypted the data in the incoming broadcast. And then saved it to Sqlite database. In the simulation system, the Nodemcu device represented the smart home device. Performance measurements of various encryption algorithms were made on the Nodemcu device. Performance measurements were carried out using Aes, one of the traditional encryption algorithms, and Ascon, Speck and Chacha algorithms, which are lightweight encryption algorithms. 128bit, 192bit and 256bit long keys were implemented separately on all encryption algorithms. Speck and Aes encryption algorithms are block cipher algorithms. Modes are among the factors that affect performance in block cipher algorithms. ECB, OFB, CFB, CTR, CBC modes have been implemented separately in block cipher operations. The study aimed to obtain the performance differences between lightweight encryption algorithms and traditional encryption algorithms. In addition, it is aimed to determine the effects of key lengths and modes on encryption algorithms.

1. GİRİŞ

Evler, insan yaşamında merkezi bir rol oynar ve birçok açıdan önemlidir. Evlerin insan yaşamına çeşitli alanda faydaları bulunur. İnsanları dış etkenlerden, hava koşulları, hayvanlar ve diğer tehlikelerden koruma görevi vardır. İnsanların dinlenmek, uyumak ve günlük yaşamlarını sürdürmeleri için kişisel alan sağlar. Aile üyeleri arasında iletişimi destekler ve aidiyet duygusunu yaratır (Eray, 2015).

Modern yaşamın gereksinimlerine uyum sağlamak ve yaşam kalitesini artırmak için akıllı evlerde yaşam artmıştır. Akıllı evlerde yaşamın tercihinde çeşitli sebepler olsa da birkaç sebep öne çıkmaktadır. Akıllı evlerin geleneksel evlere oranla daha fazla teknolojik imkana sahip olmasıdır. Teknolojik imkanlar sayesinde ev sahipleri, evdeki aydınlatma, ısıtma-soğutma, güvenlik ve diğer kontrol sistemlerini uzaktan kontrol etme imkanına sahip olurlar. Akıllı evler güvenliği ve emniyeti sağlamak için hareket sensörleri, kamera izleme ve alarm sistemlerini kullanır. Akıllı evlerde, enerji verimliliğini arttırmak için akıllı termostat ve gelişmiş enerji yönetim sistemleri kullanılır (Koçyiğit, 2020). Akıllı evleri geleneksel evlerden ayıran bir diğer özellik ise sağladığı kolaylıklardır. Otomatik ışık ayarı, perde ayarı ve sesli komutlar ile ev cihazlarını yönetme imkanları akıllı evlerin geleneksel evlere oranla avantajlı taraflarıdır.

Akıllı ev sistemlerinde, kullanıcılar akıllı ev cihazlarını uzaktan yönetebilme imkanına sahiplerdir. Bu imkân sayesinde evde olmadıkları anlarda ya da örneğin, ev ışıklarını kapatıp açma, iklimlendirme sistemini yönetme, güvenlik kameralarını izleme gibi eylemleri gerçekleştirebilmektedirler. Kullanıcılar, akıllı ev cihazlarını internet ya da yerel ağ üzerinden kontrol edebilmektedirler. Kullanılan bu ağ üzerinden kullanıcı ve ev hakkında kritik bilgiler taşınabilmektedir. Taşınan verilere örnek olarak, kullanıcının sistem yönetiminde kullandığı kullanıcı adı parola, güvenlik kameralarının kontrol komutları gibi bilgiler verilebilir (Yumurtacı ve ark, 2009).

Ağ üzerinde taşınan verilerin güvenliği için veri şifreleme algoritmaları tercih edilmektedir. Akıllı ev cihazlarının da şifreleme algoritması olarak hafif sıklet algoritmaları ön plandadır. Hafif sıklet şifreleme algoritmalarının akıllı ev cihazlarında

tercih edilme sebepleri cihazlarda uzun kullanımda performans düşüklüğüne neden olmaması ve enerji tüketimde artışa sebebiyet vermemeleri gösterilebilir. Hafif sıklet şifreleme algoritmalarının diğer şifreleme algoritmalarında ayıran düşük karmaşıklık seviyesidir. Bu sayede düşük işlem gücüne sahip IoT cihazlarında performans kaybı olmaz ve güç tüketimi düşük olur. Pil veya batarya ile çalışan akıllı ev cihazları için düşük güç tüketimi önemli kriterlerden biridir.

1.1. Tezin Kapsamı

Akıllı ev sistemlerinde kullanılan cihazlar ve iletişim ağı simüle edilmiştir. Tezin kapsamı, hafif sıklet şifreleme algoritmalarının akıllı ev cihazları üzerinde uygulanması ve performans analizi yapılmasıdır.

1.2. Tezin Amacı

Akıllı ev sistemleri Nodemcu ve Raspberry Pi3 cihazları ile modellenmiştir. Akıllı ev sistemleri modellenirken sistemi oluşturan üç temel kısıma dikkat edilmiştir. Merkezi kontrol birimi, algılayıcılar ve aktüatörler akıllı ev sistemlerinin temel bileşenleridir. Oluşturulan sistemde Nodemcu cihazı aktüatör ve algılayıcı görevlerini yerine getirir. Raspberry Pi cihazı ise merkezi kontrol birimi rolündedir. Nodemcu cihazı ile Raspberry Pi cihazı arasında kablosuz veri iletişimi kurulmuştur. Kurulan kablosuz veri iletişiminde veri hafif sıklet şifreleme algoritma türlerinden Ascon, Chacha20 ve Speck kullanılarak veri şifrelemesi yapılmıştır. Bu algoritmaların yanında simetrik şifreleme algoritma olan Aes kullanılarak veri şifreleme gerçekleştirilmiştir.

Ascon, Chacha20, Speck ve Aes şifreleme algoritmaları ile veri şifreleyerek, şifreleme algoritmalarının akıllı ev cihazları gibi sınırlı işlem gücüne sahip cihazlarda oluşturdukları performans ve enerji tüketim etkilerinin analiz edilmesi amaçlanmıştır.

1.3. Literatür Araştırması

KURU (2021) çalışmasında düşük güç tüketimi ve düşük işlem gücü olan cihazlarda Present-80, Present-128, Simon 64/96 ve Speck 64/96 hafif sıklet blok şifreleme algoritmalarının yazılım başarım analizini gerçekleştirmiştir.

ÇAVUŞOĞLU ve AL-SANABANI (2019) çalışmalarında sınırlı kaynaklara sahip IoT cihazlarda Aes, Present, LBlock, Skipjack, Simon, Xtea, Prince, Piccolo, Hight ve Rectangle gibi farklı şifreleme algoritmalarını uygulayarak enerji tüketimi, veri iletim

hızı ve şifreleme ve çözme süreleri gibi parametreler açısından test gerçekleştirmişlerdir.

PANAHI (2022) çalışmasında kablosuz algılayıcı ağlarının güvenlik taleplerini karşılamak için AES, PRESENT, LBlock, Skipjack, SIMON, XTEA, PRINCE, Piccolo, HIGHT ve RECTANGLE gibi hafif sıklet şifreleme algoritmalarını çeşitli şifre çözme modlarında performans karşılaştırması yapmışlardır.

KÜÇÜKÖMEROĞLU (2021) çalışmasında RSA, Diffie-Hellman ve ElGamal gibi asimetrik kriptografi algoritmalarının yazılım uygulamalarını incelemektedir. Kısıtlı kapasiteye sahip cihazlarda kriptografi tekniklerinin uygulamanın zor olduğunu ve performans iyileştirmesi için şifreleme ve yazılım uygulamalarının performans optimizasyonuna ihtiyaç olduğu sonucu varmıştır.

ÖZDENİZ (2023) çalışmasında, akıllı evler konusunda tasarım ve uygulama detayları ile bu süreçte karşılaşılan problemleri ve sıklıkla yapılan hatalar üzerinde durmuştur.

İLKBAHAR ve ark. (2021) çalışmalarında akıllı ev sistemlerinin bir örneği sunulmaktadır. Bu sistemde, Telegram uygulaması sayesinde kullanıcılar, evden uzaktayken bile aydınlatma, güvenlik ve giriş kontrolü gibi işlemleri gerçekleştirebilirler. Akıllı ev sistemleri üzerine bir model önerisinde bulunmuşlardır.

AVCI (2022) çalışmasında, akıllı evlerde kullanılan IoT cihazlarının ve uygulamalarının siber güvenlik açısından yaşanan sorunlar, siber saldırılar, güvenlik açıklıkları ve güvenlik açısından korunabilmek için alınması gereken önlemler incelemiştir. Ayrıca yaşanan güvenlik sorunları siber güvenlik açısından değerlendirilerek çözüm yolları önermiştir.

KANDIR ve ark. (2022) çalışmalarında ev ağı içindeki güvenlik zafiyetlerini ele almak ve özellikle UPnP açıklıklarını belirlemek amacıyla yapılan bir çalışmayı özetlemişlerdir. Ve evdeki ağ ve iletişim güvenliği yanında her bir IoT cihazının güvenliğinin önemini vurgulamışlardır.

ŞAHİN (2015) çalışmasında modern blok şifreleme algoritmalarının, özellikle AES (Advanced Encryption Standard), DES (Data Encryption Standard) ve 3DES'in günümüzde kullanılan kriptografide önemli bir rol oynadığını belirtmiştir. Bu algoritmaların güvenlik açısından önemli olduğunu vurgular ve çalışmanın bu algoritmaları inceleyerek bilgi sunmayı amaçladığını belirtmiştir.

KATAGI ve MORIAI (2008) çalışmalarında hafif kriptografide standardizasyonun öneminden de bahsetmişledir. IoT'nin büyümeye devam etmesi ile birlikte, hafif kriptografik çözümlere olan talep de artması beklenildiğine vurgu yapmışlardır. Hafif sıklet şifreleme algoritmaların sınırlı kaynaklı cihazları siber saldırılara karşı korumak için tasarlanmış verimli ve güvenli kriptografi türü olduğu belirtmişlerdir.

MUSA ve ark (2003) çalışmalarında AES algoritmasının basitleştirilmiş versiyonunu tanımlamışlardır. Basitleştirilmiş versiyonla gerçek versiyonun anlaşılmasının kolaylaştırılması amaçlanmıştır. Basitleştirilmiş versiyon kullanılarak hem doğrusal hem de farklı kriptanaliz yöntemleri gerçekleştirmişlerdir.

ÖZTÜRK ve NAIMI (2017) çalışmalarında akıllı ev sistemleri alanında kullanılan yöntemlerin farklarını, avantajlarını ve dezavantajlarını ele almaktadır. İç aydınlatma kontrolünde kullanılan PLC (Programmable Logic Controller) yöntemi ile kablosuz ağ teknoloji temelli bir akıllı ev sistemi arasındaki karşılaştırmayı içermektedir.

ASLAN ve SAKALLI (2004) çalışmalarında, AES (Advanced Encryption Standard) algoritmasının bir parçası olan S-kutularının kriptografik özelliklerini incelemektedir. AES'in anahtar bağımlılığını artırmak ve AES'in kriptografik güvenliğini artırmak için kullanılan S-kutularının tasarımı hakkında bilgi vermektedir. S-kutuları, 8-bit girdi bloklarını 8-bit çıktı bloklarına dönüştürmek için kullanılan bir tablodur. S-kutuları, AES'in Değiştirme-Karıştırma (Substitution-Permutation) tasarım temeline dayanır. Makale, iyi tasarlanmış bir S-kutusunda bulunması gereken kriptografik özellikleri sıralamaktadır

GÜĞÜL ve SARITAŞ (2011) çalışmalarında güncel hayatımızı daha emniyetli ve pratik hale getirecek bir Akıllı Ev Modeli tasarlanması ve gerçekleştirilmesi üzerine odaklanmışlardır. PIC16F877 ve PIC16F628 mikroişlemcileri akıllı ev modeli geliştirmişlerdir.

2. AKILLI EV SİSTEMLERİ VE GÜVENLİĞİ

Akıllı ev sistemleri, evin kullanım alanlarına yerleştirilen akıllı sensörler, güvenlik kameraları, hoparlörler, ampuller vb. diğer teknolojik cihazların birbirleriyle uyumlu şekilde kullanıldığı sistemlerin kurulduğu evlerdir. Bu sistemler genellikle akıllı telefonları merkezine alan, sesli asistanlarında kullanılabildiği merkezi yönetim imkânı sunar. Ayrıca akıllı ev sistemleri, bir yazılım tarafından programlanarak binaları da uzaktan kontrol etmeye yarar. Bu sistemler kablolu olabileceği gibi kablosuz da olabilir. Teknolojinin tüm artıları kullanılarak oluşturulan akıllı ev sistemleriyle ilgili birçok değişken bulunmaktadır (Korkmaz, 2017).

Akıllı ev sistemleri, evlerdeki geleneksel sistemlere göre bir dizi avantaj sunar. Akıllı ev sistemlerinin faydalarını öne çıkaran bazı unsurlar şunlardır (Akıllı Ev Sistemleri Nedir? | TeoremEnerji | 2023):

- **Konfor ve kolaylık**

Akıllı ev sistemleri, ev sahiplerine konfor ve kolaylık sağlar. Uzaktan erişim imkânı sunarak, kullanıcılar evlerindeki aygıtları, ışıkları, termostatları ve güvenlik sistemlerini tek bir merkezi kontrol noktasından yönetebilirler.

- **Enerji verimliliği**

Akıllı termostatlar, enerji verimli aydınlatma sistemleri ve enerji kullanımını izleyen sensörler sayesinde, akıllı ev sistemleri evlerin enerji tüketimini optimize edebilir. Bu da enerji tasarrufu ve daha düşük enerji faturaları anlamına gelir.

- **Güvenlik artışı**

Akıllı ev sistemleri, güvenlik kameraları, hareket sensörleri, akıllı kilitler ve kapı zilleri gibi unsurlarla ev güvenliğini artırır. Kullanıcılar, akıllı telefonları aracılığıyla evlerini izleyebilir ve kontrol edebilirler.

- **Ev otomasyonu**

Akıllı ev sistemleri, ev sahiplerine çeşitli senaryolar ve otomasyon seçenekleri sunar. Örneğin, bir ev sahibi eve gelir gelmez ışıkları açabilir, termostatı ayarlayabilir ve güvenlik sistemi etkinleştirilebilir.

- **Uzaktan izleme ve kontrol**

Ev sahipleri, akıllı ev uygulamalarını kullanarak evlerini uzaktan izleyebilir ve kontrol edebilirler. Bu, seyahat edenler için güvenlik ve rahatlık sağlar.

- **Yaşam kalitesi artışı**

Akıllı ev sistemleri, yaşlı bireyler veya engelliler için yaşam kalitesini artırabilir. Otomasyon ve uzaktan kontrol, günlük yaşamı daha erişilebilir hale getirebilir.

- **Eğlence ve ses kontrolü**

Akıllı ev sistemleri, eğlence sistemlerini (TV, ses sistemleri) entegre ederek kullanıcıların konforunu artırır. Ses kontrolleri aracılığıyla ev sahipleri, cihazları sesli komutlarla yönetebilirler.

Bu faydalar, akıllı ev sistemlerinin giderek daha popüler hale gelmesine ve ev sahiplerinin yaşamlarını daha pratik, güvenli ve verimli hale getirmelerine katkıda bulunmaktadır.

2.1. Akıllı Ev Sistem Mimarileri

Akıllı ev sistemleri, farklı mimariler ve bileşenlerle tasarlanabilmektedir. Bazı akıllı ev sistemleri mimarilerini şu şekilde listeleyebiliriz (Inohom Akıllı Ev Sistemleri, 2023):

- **Merkezi kontrol mimarisi**

Tüm akıllı cihazlar merkezi bir kontrol ünitesi tarafından yönetilir. Sensörler, ışıklar, ısıtma-soğutma sistemleri, güvenlik kameraları ve diğer cihazlar bu merkezi üniteye bağlıdır. Kullanıcılar, akıllı telefon veya tablet aracılığıyla bu merkezi üniteyi kullanarak evlerini yönetebilirler.

- **Dağıtılmış mimariler**

Her oda veya bölüm için ayrı bir kontrol ünitesi bulunur. Her odanın ihtiyacına göre sensörler ve cihazlar yerleştirilir. Bu yöntem, daha özelleştirilmiş ve esnek bir kontrol sağlar.

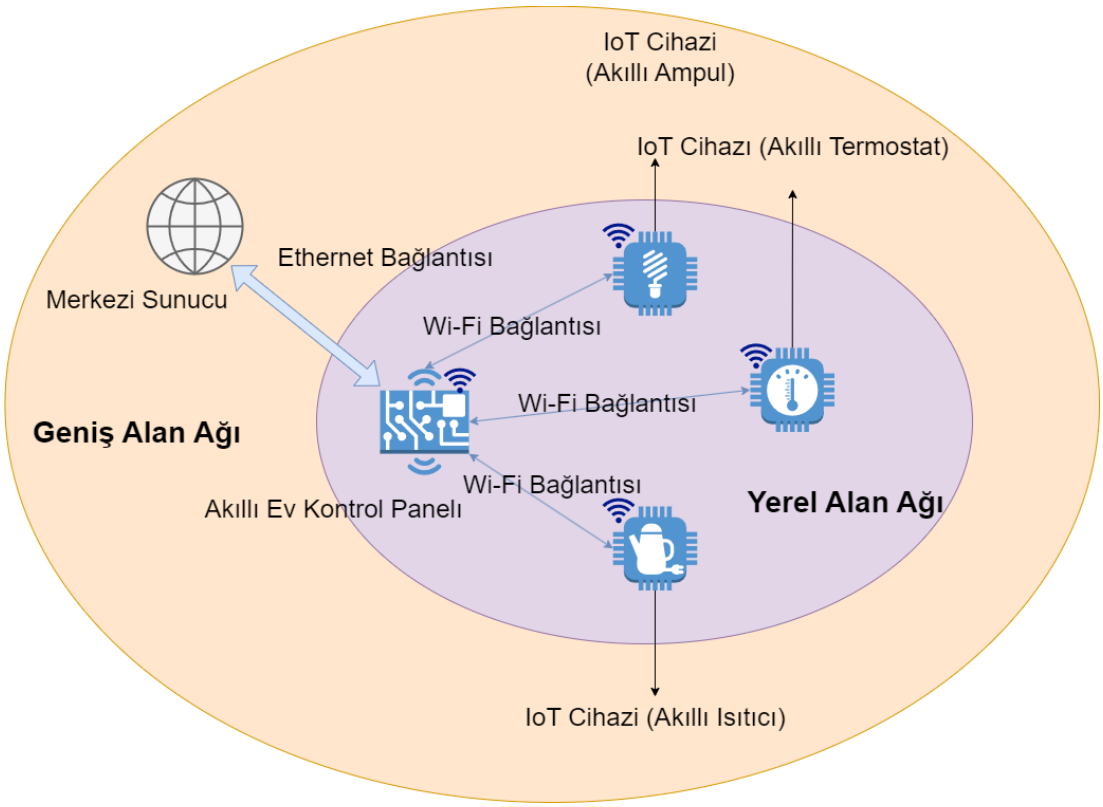
- **Bulut tabanlı mimariler**

Akıllı ev verileri bulut tabanlı bir hizmette depolanır ve yönetilir. Kullanıcılar, internet üzerinden bu bulut hizmetine erişerek evlerini kontrol edebilirler. Bu yöntem, uzaktan erişim ve veri paylaşımı için uygundur.

- **Karma mimariler**

Merkezi kontrol ünitesi ile dağıtılmış kontrol üniteleri bir arada kullanılır. Örneğin, merkezi ünite ısıtma-soğutma sistemini yönetirken, odalardaki sensörler ayrı ayrı aydınlatma ve güvenlik sistemlerini kontrol edebilir.

Akıllı ev sistemleri, kullanıcıların ihtiyaçlarına ve tercihlerine göre farklı mimarilerle tasarlanabilmektedir (Mesleğim Hayatım, 2020).



Şekil 2.1. Örnek Akıllı Ev Sistem Mimarisi

Akıllı ev cihazları, IoT cihazlarının özel amaçlarda kullanılmak üzeri tasarlanmış versiyonlarıdır. Akıllı ev cihazların işlemci, sensörler ve aktüatörler gibi temel parçaları vardır. Sensörler çevreden gelen bilgileri işlemciye gönderir. Aktüatörler ise çevreye işlemciden gelen veriler doğrultusunda etki eder.

Akıllı ev sistemlerinde, akıllı cihazlar bilgi üretir. Bilgi üretimi, sensörlerinden gelen verileri işleyerek olabilir. Aktüatörlerin işlemi sonucunda cihaz yeni bilgi üretmiş olur.

Şemada görüldüğü gibi akıllı cihazlar kablosuz ağ bağlantısına sahiptir. Akıllı ev kontrol birimi de kablosuz ağ bağlantısına sahiptir. Üç akıllı cihaz kablosuz ağ bağlantısı ile akıllı ev kontrol paneline bağlanmıştır. Cihazların bağlantısı yerel üzerinden gerçekleşmektedir.

Akıllı cihazların arasından akıllı termostat üzerinde örnek olarak ele alınırsa akıllı termostat evin sıcaklık durumunu takip eden ve belirli sıcaklık değerlerinde aksiyon alan cihazdır. Akıllı cihazdaki sensörler sayesinde çevre sıcaklığı ölçebilmektedir. Aktüatör sayesinde iklimlendirme sistemine emir verebilmektedir.

Akıllı termostatın ürettiği çeşitli bilgiler vardır. Belli bir zaman dilimi göz önüne alınacak olunursa o anda evin kaç derece olduğu bilgisi ve iklimlendirme sisteminin aktif ya da pasif durumda olma bilgisi akıllı termostata bulunmaktadır. Akıllı termostatın üretmiş olduğu bu bilgileri hali hazırda bağlı olduğu yerel ağdaki kontrol merkezine göndermesi gerekmektedir.

Kullanıcılar HMI (Human Machine Interface)'ler ile kontrol merkezini kullanabilmektedir. Kontrol merkezinde, kontrol merkezinin bağlı olduğu cihazlar hakkında bilgiler elde edip onların kontrolünü sağlayabilmektedirler.

Akıllı cihazlar ile kontrol biriminin arasında kullanıcıya ait bilgilerin iletimi gerçekleşir. İletişim esnasında taşınan verinin gizlilik açısından şifrelenmiş olması gerekmektedir. Akıllı cihazlar güç verimliliği ön planda tutularak geliştirilen cihazlar oldukları için düşük işlem gücüne sahiptirler. Bu yüzden geleneksel şifreleme algoritmalarını verimli şekilde çalıştıramazlar. Düşük güçteki cihazlar için geliştirilmiş daha az karmaşıklığa sahip şifreleme algoritmaları vardır. Bu algoritmaları hafif sıklet şifreleme algoritmalar adı altında ele alınır.

2.2. Akıllı Ev Sistemi Unsurları

Akıllı ev sistemleri mimarisinde kullanılan temel unsurlar şunlardır:

- **Merkezi kontrol birimi (hub/controller)**

Akıllı ev sisteminin beyni olarak görev yapar. Bu birim, diğer akıllı cihazlarla iletişim kurar, verileri işler ve ev sahibinin tercihlerine göre otomasyonları yönetir.

- **Sensörler**

Akıllı ev sistemleri, çeşitli sensörleri kullanarak çevresel değişiklikleri algılar. Örneğin, hareket sensörleri, kapı ve pencere sensörleri, ısı ve ışık sensörleri gibi sensörler kullanılabilir.

- **Aygıtlar ve aktüatörler**

Aygıtlar, akıllı ev sistemine bağlı cihazları temsil eder. Işıklar, termostatlar, güvenlik kameraları, kapı kilitleri, prizler ve daha fazlası bu kategoriye girer. Aktüatörler ise belirli bir eylemi gerçekleştiren cihazları ifade eder.

- **Ağ altyapısı**

Akıllı ev cihazları arasındaki iletişimi sağlamak için bir ağ altyapısı gereklidir. Bu genellikle Wi-Fi, Zigbee, Z-Wave gibi kablosuz iletişim protokollerini içerir.

- **Veri depolama**

Akıllı ev sistemleri, kullanıcı tercihleri, otomasyon senaryoları ve cihazların geçmiş durumları gibi verileri depolamak için bir veri tabanına ihtiyaç duyar.

- **Güvenlik protokolleri**

Akıllı ev sistemlerinin güvenliği büyük önem taşır. Şifreleme, güvenlik kameraları, güvenlik sensörleri ve güvenlik protokollerinin kullanımı gibi önlemler, ev sahiplerinin güvenliğini sağlamaya yardımcı olur.

- **Uygulama ve kullanıcı arabirimi**

Kullanıcılar genellikle bir akıllı ev uygulaması veya web tabanlı bir kullanıcı arayüzü aracılığıyla sistemi kontrol ederler. Bu uygulamalar, aygıtları yönetme, otomasyon senaryoları oluşturma ve evin durumu hakkında bilgi almayı sağlar.

- **Enerji yönetimi**

Akıllı ev sistemleri, enerji verimliliği sağlamak için termostatlar, enerji izleme cihazları ve enerji yönetimi özelliklerini içerebilir.

2.3. Akıllı Ev Sistemlerinde Veri İletim Yöntemleri

Akıllı evler, cihazların birbirleriyle iletişim kurabildiği ve ev sahiplerinin bu cihazları uzaktan kontrol edebildiği evlerdir. Cihazlar birbirleri arasında veri iletiminde çeşitli ağ teknolojileri kullanırlar. Kullanılan ağ teknolojileri iletişim sağlayan cihazların özelliklerine, ortam şartlarına ve kullanım senaryosuna göre değişiklik gösterir. Veri iletiminde yaygın olarak kullanılan ağ teknolojilerine Wifi, Bluetooth, Zigbee, Z-Wave, RFID, NFC, LoRa örnek verilebilir.

2.3.1. Wi-Fi

IoT cihazları internete bağlanmak için birçok farklı teknoloji kullanabilirler. Wi-Fi en yaygın kullanılan teknolojilerden biridir (Nesnelerin İnterneti (IoT) Nedir? 2020). Wi-Fi, IEEE 802.11a, IEEE 802.11b, IEEE 802.11g, IEEE 802.11n ve IEEE 802.11ac standartlarına göre belirlenir 1. Wi-Fi, kişisel bilgisayar, akıllı telefon, oyun konsolu ve dijital işitsel cihazların kablosuz ağ sahası içerisinde internete bağlanmasını sağlayan bir teknolojidir 12. Wi-Fi'nin erişim noktaları, bir oda gibi küçük alanlardan, birkaç yüz metrekarelik geniş alanlara kadar etki edebilir, buralarda internet erişimini sağlayabilirler 1. Wi-Fi, yüksek hızlı veri transferi, geniş kapsama alanı ve düşük maliyetli olması nedeniyle IoT cihazları için en yaygın kullanılan teknolojilerden biridir. (WiFİ Nedir? Nasıl Çalışır? Özellikleri Nelerdir? - Elektrikde, 2021) Wi-Fi RTLS (Real-Time Location System), konum takibi ve yer belirleme sistemleri için de kullanılabilir 1. Wi-Fi, WEP, WPA ve benzeri kablosuz şifreleme yöntemleri veya IEEE 802.1x gibi yetkilendirme yöntemleriyle çeşitli güvenlik seçenekleri sunar 1. Wi-Fi, lisans gerektirmeyen frekanslarda çalışır ve ağ için kablolama gereksinimi yoktur, böylece kablo çekilemeyecek binalarda veya binalar arası bağlantılarda kolaylık sağlar 1. Wi-Fi, dünya genelinde kullanılan bir standart kümesidir ve dünyanın her yerinde aynı şekilde çalışır (Nesnelerin İnterneti Nedir? Tanım ve Açıklama, 2023).

2.3.2. Bluetooth

Bluetooth, kablosuz bir bağlantı teknolojisidir ve mobil cihazlar arasında ses ve veri haberleşmesini sağlar. Bluetooth, yakın mesafelerde işe yarayan bir teknolojidir ve 10 ila 100 metre arası bir mesafeden mobil veya sabit cihazlarının birbirleri arasında veri aktarımı yapmaya veya kişisel alan ağları (PAN) kurmaya yarayan kablosuz bir bağlantı standardıdır. IoT cihazlarında tercih edilme nedeni, Bluetooth'un düşük güç tüketimi, düşük maliyeti ve kablosuz bağlantı sağlamasıdır.

2.3.3. RFID

RFID (Radyo Frekans Tanımlama), bir nesnenin kimliğini tanımlamak için kullanılan bir teknolojidir. RFID etiketleri, bir radyo dalgası aracılığıyla veri okuyucularına iletilen dijital verileri içerir. Bu teknoloji, IoT cihazlarında kullanıldığında, cihazların birbirleriyle ve internetle iletişim kurmasını sağlar (RFID Nedir? RFID Etiketlerinin Avantajları Nelerdir? | ETMD, 2022). RFID etiketleri, benzersiz bir kimlik numarası taşıdığından, envanter yönetimi, üretim takibi, malzeme yönetimi, tedarik zinciri yönetimi, güvenlik, erişim kontrolü, hayvan takibi, kütüphane yönetimi, hastane yönetimi, araç takibi ve daha birçok alanda kullanılabilir (İlgazi, 2022). RFID etiketlerinin avantajları arasında hızlı ve kağıtsız veri toplama, barkodlara nazaran daha fazla verinin tutulabilmesi, mevzuata uygunluk, ekipman denetimlerinin, bakım işleri vb. yönetilmesi, nemli, tozlu, kirli koşullar, aşındırıcı ortamlar, titreşim ve darbe gibi en zorlu ortamlarda güvenilir operasyon, temasa ve gözlemeye gerek yoktur (Nesnelerin İnterneti (IoT) Nedir? Neden Önemlidir? ; Midas, 2022).

2.3.4. NFC

NFC (Yakın Alan İletişimi), IoT cihazları için önemli bir teknolojidir. IoT cihazları, fiziksel dünyadaki nesnelerin internete bağlanmasını sağlayan cihazlardır. NFC, IoT cihazlarının birbirini tanımasını, verileri paylaşmasını ve birlikte çalışmasını sağlar (NFC Nedir, Nasıl Kullanılır? 2021). NFC teknolojisi, IoT cihazları arasında veri güvenliğini sağlayarak yüksek güvenlikli kriptografi desteği sunmaktadır. Ayrıca, NFC teknolojisi, IoT uygulamalarının genişlemesine yardımcı olmuş ve evdeki akıllı cihazların basit bir dokunuşla iletişim kurmasını sağlamıştır. NFC'nin cihazlar arasında hızlı ve basit bir şekilde iletişim kurabilme özelliği, IoT cihazlarının etkileşimini kolaylaştırarak akıllı ev sistemlerinin ve diğer IoT uygulamalarının daha verimli çalışmasına olanak tanımaktadır (NFC ve IoT: Nesnelerin İnternetine Entegrasyon; Avukat Kartviziti, 2020).

2.3.5. Zigbee

Zigbee, düşük güç tüketimi gerektiren cihazlar arasında kablosuz bağlantı sağlayan bir iletişim protokolüdür. Zigbee, Wi-Fi ve Bluetooth'a göre daha az bant genişliği gerektirir ve akıllı ev sensörleri gibi düşük güçlü cihazlar için daha uygun bir seçenek olarak düşünülmektedir. Zigbee, farklı üreticilerden cihazların birbirleriyle konuşmasını sağlayan bir ortak dil kullanır. Bu sayede, akıllı ev cihazlarının birbirleriyle konuşabilmesi ve merkezi bir kontrolör olmadan bir arada çalışabilmesi mümkün hale gelir. Zigbee, akıllı evlerin daha verimli ve daha uyumlu hale gelmesine yardımcı olur.

IOT cihazları, internete bağlı cihazlar olarak tanımlanabilir. Zigbee, IOT cihazları için önemlidir çünkü düşük güç tüketimi gerektiren cihazlar arasında kablosuz bağlantı sağlar. Zigbee, akıllı evlerin daha verimli ve daha uyumlu hale gelmesine yardımcı olur. Zigbee protokolü, akıllı ev cihazlarının birbirleriyle konuşabilmesini ve merkezi bir kontrolör olmadan bir arada çalışabilmesini mümkün kılar. Bu sayede, akıllı ev cihazlarının birbirleriyle konuşabilmesi ve merkezi bir kontrolör olmadan bir arada çalışabilmesi mümkün hale gelir.

2.3.6. Z-Wave

Z-Wave, kablosuz bir ağ teknolojisidir ve Zigbee gibi akıllı ev cihazları için önemlidir. Z-Wave, akıllı ev cihazlarının birbirleriyle konuşabilmesini ve merkezi bir kontrolör olmadan bir arada çalışabilmesini mümkün kılar. Z-Wave, Zigbee'den daha az bant genişliği gerektirir ve akıllı ev sensörleri gibi düşük güçlü cihazlar için daha uygun bir seçenek olarak düşünülmektedir. Z-Wave, akıllı evlerin daha verimli ve daha uyumlu hale gelmesine yardımcı olur (Z-Wave ve ZigBee Nedir? Nasıl Seçim Yapılmalıdır? 2019).

2.3.7. LoRa

LoRa, düşük güç tüketimli kablosuz haberleşme teknolojisidir ve IoT cihazları için önemlidir. LoRa, düşük güç tüketimli uzun mesafeli haberleşme sağlar ve IoT cihazlarının birbirleriyle konuşabilmesini ve merkezi bir kontrolör olmadan bir arada çalışabilmesini mümkün kılar. LoRa, LPWAN (düşük güçlü geniş alan ağı) nesnelerin interneti (IoT) için bir ağ katmanı teknolojisidir (Lora Teknolojisi Nedir?| Nerelerde Kullanılır ; Esgsistem, 2020). LoRa, düşük güç tüketimi, uzun menzilli iletim, düşük bant genişliği, basit ağ yapısı ve düşük işletme maliyetleri gibi avantajlara sahiptir.

LoRa, IoT cihazlarının birbirleriyle konuşabilmesini ve merkezi bir kontrolör olmadan bir arada çalışabilmesini mümkün kılar. LoRa, akıllı şehirler, akıllı tarım, akıllı evler, akıllı binalar, akıllı ulaşım ve endüstriyel IoT gibi birçok alanda kullanılmaktadır (Akıllı Ev Teknolojileri- INTERPOINT Teknoloji, 2018) .

2.4. Veri Şifreleme

Akıllı ev sistemleri, evimizi uzaktan kontrol etmemizi sağlayan cihazların ve hizmetlerin bir kombinasyonudur. Bu cihazlar, ışıkları, termostatları, kilitleri, kameraları ve daha fazlasını kontrol edebilir.

Akıllı ev sistemleri, evimizi daha rahat, verimli ve güvenli hale getirebilir. Ancak, bu sistemler aynı zamanda veri güvenliği riskleri de oluşturabilir. Akıllı ev sistemlerinde veri güvenliği tehdit eden faktörlere kötü amaçlı yazılımları, erişim ihlallerini ve zayıf şifreleri verebiliriz. Kullanıcıların akıllı ev sistemlerinde verilerinin güvenliği sağlayabilmek için çeşitli aksiyonlar almalıdırlar. Akıl ev sisteminde kullandıkları cihazların yazılımlarını ve cihazlarla etkileşim sağlamak için kullandıkları mobil uygulamalarını güncel tutmaları gerekmektedir. Cihazların yönetim paneline girişte kullandıkları şifreleri güçlü şifreler ile değiştirmeleri gerekmektedir. Cihazlara erişim için destekleyen cihazlarda iki faktörlü doğrulama kullanılmalıdır. Kullanıcılar bu işlemler haricinde cihazların anlık durum izleme arayüzlerinden sistemi takip etmeleri ve olağan dışı bir durum olup olmadığı hakkında bilgi edinmeleri gerekmektedir.

Kullanıcı ile akıllı ev cihazları arasındaki veri iletişim güvenliğinin öneminin yanında akıllı ev sistemindeki cihazların birbirleri arasındaki veri iletişim güvenliği de önemlidir. IoT cihazları, internete bağlı cihazlar ve ağlar arasında veri paylaşımı yaparlar. Bu nedenle, IoT cihazları arasında veri güvenliği büyük önem taşır. Veri şifreleme, IoT cihazları arasında veri iletiminde önemli bir rol oynar. Veri şifreleme, verilerin sadece yetkili kullanıcılar tarafından okunabilmesini sağlar. Bu, verilerin siber saldırılara karşı korunmasına yardımcı olur. Veri şifreleme, IoT cihazlarındaki güvenlik açıklarını azaltmaya yardımcı olabilir. Ancak, veri şifreleme, IoT cihazlarının performansını da etkileyebilir. Bu nedenle, IoT cihazlarındaki veri şifreleme, performans ve güvenlik arasında bir denge kurmak için dikkatli bir şekilde yapılmalıdır (Iot'de Veri Tazeliğinin Önemi | Ankaref, 2020).

Akıllı evler, evdeki cihazların birbirleriyle iletişim kurmasını ve kablosuz ağlara bağlı bu cihazların tüketiciler hakkında veri toplamasını mümkün kılar. Bu nedenle, akıllı

evlerde veri güvenliği oldukça önemlidir (Evlerde Kullanılan Akıllı Cihazlar Ciddi Güvenlik Riskleri Taşıyor Olabilir, 2023). Kötü niyetli kişiler, akıllı ev aletleri vasıtasıyla mahrem bilgileri elde edip fidye yazılım sistemleriyle şantaj yapabilirler (Colak, 2020). Akıllı ev sistemleri, kişinin alışkanlıkları profilleme yapılarak birtakım etik problemlere neden olabilir. Bu nedenle, akıllı evlerde veri güvenliği sağlamak için, cihazların güvenliği çok önemlidir. Örneğin, akıllı ev sistemlerinin şifreleri güçlü olmalı, güncellemeleri düzenli olarak yapılmalı ve cihazların varsayılan ayarları değiştirilmelidir.

Veri şifreleme, verilerin yetkisiz erişimden korunmasını sağlar ve veri güvenliğini artırır. Şifreleme, verilerin yalnızca belirli bir kodun (anahtar) çözebileceği veri karıştırmadan kaynaklanır. Basitçe söylemek gerekirse, insan tarafından okunabilen metni anlaşılmaz ifadelere, yani şifreli metne dönüştürür. Veri şifrelemenin önemi, siyah şapkaların süregelen veri hırsızlığı tehdidini dayatmasından kaynaklanmaktadır. Çözüm olarak, modern güvenlik araçları ve şifreleme yöntemleri, en üst düzeyde gizlilik koruması sağlar (Şifreleme Nedir ve Siber Güvenlik İçin Neden Önemlidir? 2023). Veri şifreleme, kişisel bilgilerin gizliliğini sağlar ve verilerin bütünlüğünü korur.

Geleneksel şifreleme algoritmaları, yüksek güvenlik seviyeleri sağlamak için tasarlanmıştır. Bu algoritmalar, genellikle daha büyük anahtar boyutlarına ve daha karmaşık işlemlere sahiptir. Hafif sıklet şifreleme algoritmaları ise, daha küçük anahtar boyutlarına ve daha az karmaşık işlemlere sahiptir. Bu nedenle, hafif sıklet şifreleme algoritmaları, daha az kaynak tüketir ve daha hızlıdır. Hafif sıklet şifreleme algoritmaları, özellikle IoT cihazları gibi kaynakları sınırlı cihazlar için tasarlanmıştır. Bu cihazlar, daha küçük anahtar boyutlarına ve daha az karmaşık işlemlere sahip olacak şekilde tasarlanmıştır. Geleneksel şifreleme algoritmaları, daha yüksek güvenlik seviyeleri sağlar, ancak daha fazla kaynak tüketirler. Hafif sıklet şifreleme algoritmaları, daha az kaynak tüketir, ancak daha düşük güvenlik seviyeleri sağlarlar.

Geleneksel şifreleme algoritmaları akıllı ev cihazları için elverişli değildirler. Akıllı ev cihazları, geleneksel şifreleme yöntemlerini kullanmazlar çünkü bu yöntemler, daha büyük anahtar boyutlarına ve daha karmaşık işlemlere sahiptir. Bu nedenle, geleneksel şifreleme yöntemleri, daha fazla kaynak tüketirler ve daha yavaştırlar. Hafif sıklet şifreleme algoritmaları ise, daha küçük anahtar boyutlarına ve daha az karmaşık işlemlere sahiptir. Bu nedenle, hafif sıklet şifreleme algoritmaları, daha az kaynak

tüketir ve daha hızlıdır. Hafif sıklet şifreleme algoritmaları, özellikle IoT cihazları gibi kaynakları sınırlı cihazlar için tasarlanmıştır. Bu cihazlar, daha küçük anahtar boyutlarına ve daha az karmaşık işlemlere sahip olacak şekilde tasarlanmıştır (IoT Güvenliği Ev Ağınız İçin Neden Önemlidir?, 2023). Geleneksel şifreleme algoritmaları, daha yüksek güvenlik seviyeleri sağlar, ancak daha fazla kaynak tüketirler. Hafif sıklet şifreleme algoritmaları, daha az kaynak tüketir, ancak daha düşük güvenlik seviyeleri sağlarlar.

2.4.1. Şifreleme algoritmaları

Günümüzde şifreleme algoritmaları, hafif şifreleme algoritmaları ve geleneksel şifreleme algoritmaları olmak üzere iki geniş kategoride incelenebilir. Hafif şifreleme algoritmaları, genellikle sınırlı kaynaklara sahip olan IoT (Nesnelerin İnterneti) cihazları gibi ortamlarda kullanılmak üzere tasarlanmıştır. Bu algoritmalar, düşük güç tüketimi, hızlı işleme ve sınırlı bellek gereksinimleri gibi özelliklere odaklanarak, veri güvenliğini sağlarken aynı zamanda sistem kaynaklarını verimli bir şekilde kullanmaya yönelik tasarlanır.

Diğer yandan, geleneksel şifreleme algoritmaları, geniş bir uygulama yelpazesi için daha genel amaçlı olarak tasarlanmıştır. Bu algoritmalar genellikle daha karmaşık matematiksel işlemler içerir ve geniş veri setleri üzerinde güvenli bir şekilde çalışabilirler. Örneğin, AES (Advanced Encryption Standard) ve DES (Data Encryption Standard) gibi popüler simetrik şifreleme algoritmaları, geniş çapta kabul görmüş ve yaygın olarak kullanılan geleneksel şifreleme yöntemleridir.

Şifreleme algoritmaları konusunda kullanılan terimler arasında, anahtar uzunluğu, blok boyutu, saldırı dayanıklılığı ve karmaşıklık gibi kavramlar önemlidir. Anahtar uzunluğu, kullanılan şifreleme algoritmasının güvenliği üzerinde doğrudan bir etkiye sahiptir. Genellikle daha uzun anahtarlar, daha güvenli bir şifreleme sağlar, ancak bu durumda işlemler daha yavaş olabilir. Blok boyutu, veri bloklarının işlenme şeklini ifade ederken, saldırı dayanıklılığı algoritmanın güvenlik seviyesini belirtir. Ancak işleme süresini artırabilir. Karmaşıklık ise algoritmanın işlemlerinin ne kadar karmaşık olduğunu ifade eder. İyi bir şifreleme algoritması, güvenlik gereksinimlerine uygun olarak seçilen bu terimleri dengeli bir şekilde ele almalıdır.

Şifreleme algoritmaları kullandıkları anahtar türüne göre simetrik ve asimetrik olmak üzre ikiye ayrılır.

2.4.1.1. Simetrik şifreleme

Simetrik şifreleme, aynı anahtarın hem şifreleme (veriyi şifreleme) hem de deşifreleme (şifrelenmiş veriyi orijinal haline getirme) işlemlerinde kullanıldığı bir şifreleme yöntemidir. Yani, iki taraf da aynı anahtarı bilir ve bu anahtar üzerinden iletişim kurarlar.

Şekil 2.2. Simetrik Şifreleme Algoritması

Temel prensip, şifreleme işleminde kullanılan anahtarın, deşifreleme işlemi için de kullanılabilmesidir. Bu nedenle simetrik şifreleme, verinin güvenli bir şekilde iletilmesini sağlarken, aynı zamanda işlem sürelerini hızlandırır, çünkü genellikle daha az işlem gücü gerektirir (ÖFC, 2021).

Simetrik şifreleme kullanıldığında, iki tarafın arasında güvenli bir şekilde anahtarın paylaşılması önemlidir. Anahtarın güvenli bir şekilde paylaşılmadığı durumlarda, anahtarın ele geçirilmesi durumunda şifreleme güvenliği tehlikeye girebilir. Bu sebeple anahtar yönetimi, simetrik şifreleme sistemlerinde kritik bir öneme sahiptir.

AES (Advanced Encryption Standard) gibi simetrik şifreleme algoritmaları, günümüzde geniş çapta kullanılan güvenli şifreleme yöntemleridir. Simetrik şifreleme, özellikle veri iletimi sırasında yüksek hız ve etkinlik gerektiren uygulamalarda tercih edilir.

Simetrik şifreleme algoritmaları karakter tabanlı ve bit tabanlı olmak üzere ikiye ayrılır.

- **Karakter tabanlı şifreleme**

Karakter tabanlı şifreleme, veriyi şifrelemek ve şifrelenmiş veriyi geri çözmek için kullanılan bir şifreleme yöntemidir. Bu şifreleme yöntemi, metin veya karakter dizilerini temel alır ve genellikle bir anahtar kullanılarak işlem yapar.

Karakter tabanlı şifreleme algoritmalarına Caesar, Vigenere, Base64 ve Rot13 örnek verilebilir.

- **Bit tabanlı şifreleme**

Bit tabanlı şifreleme, veriyi şifrelemek ve şifrelenmiş veriyi çözmek için bireylerin kullanılan anahtarı paylaşması gereken bir şifreleme yöntemidir. Bu yöntem, her bir bitin (binary digit) manipülasyonu üzerine kurulu olan ve genellikle sayısal verileri işleyen algoritmaları içerir.

Bit tabanlı şifreleme algoritmalarına AES, DES örnek verilebilir.

2.4.1.2. Asimetrik şifreleme

Asimetrik şifreleme, bir çift anahtarın kullanıldığı bir şifreleme yöntemidir. Bu anahtarlar genellikle "genel anahtar" (public key) ve "özel anahtar" (private key) olarak adlandırılır. İki anahtar arasındaki temel fark, genel anahtarın herkes tarafından bilinebilen bir anahtar olması, özel anahtarın ise sadece anahtar sahibi tarafından bilinen gizli bir anahtar olmasıdır. Asimetrik şifreleme, genellikle güvenli iletişim kurmak ve dijital imza gibi güvenlik uygulamalarında kullanılır.

Asimetrik şifreleme algoritmalarına RSA, ECC, Diffie-Helmen ve ElGamal örnek olarak verilebilir.

Şekil 2.3. Asimetrik Şifreleme Algoritması

2.4.2. Terminoloji

S-box (Substitution box), şifreleme algoritmalarında kullanılan bir yapıdır ve genellikle blok şifreleme algoritmalarında anahtarlar veya veri blokları üzerinde substition (yerine koyma) işlemi uygular. S-box, bir giriş setini belirli bir çıkış setine dönüştürerek veriyi karıştıran ve şifreleyen bir matematiksel fonksiyondur (S-Kutuları (S-Boxes) – Bilgisayar Kavramları, 2008). Bu yapı, şifreleme algoritmalarının güvenlik özelliklerini artırmak ve doğrusal kriptanaliz gibi saldırılara karşı direnç sağlamak amacıyla kullanılır.

Temel olarak, S-box bir substitütör veya değiştirici olarak düşünülebilir. Girişe bir değer girildiğinde, S-box belirli bir matematiksel formül veya tabloya göre bir çıkış değeri üretir. Bu çıkış değeri genellikle giriş değerinden tamamen farklıdır, bu nedenle S-box'lar, şifreleme algoritmalarının karışıklık ve güvenlik özelliklerini artırmak için kullanılır.

C (mn)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	FA	A6	A9	E5	DE	5A	05	FB	5C	AA	64	CB	A1	87	6A	6C
1	E4	87	93	6A	76	E5	66	09	43	0C	91	92	03	8F	63	30
2	54	99	E9	30	EE	BF	F2	E6	71	4E	90	D5	18	85	45	FA
3	7E	73	0E	13	8B	5B	08	8B	C8	3B	6A	10	87	09	FB	47
4	28	AF	C5	20	0B	8D	74	D5	59	37	19	C9	2A	4F	02	C1
5	91	F1	50	83	9B	42	87	4A	42	F2	74	0C	4F	2D	49	AE
6	DA	25	64	58	CD	FE	1B	D2	7D	F8	66	A8	6D	2A	A9	7E
7	21	C3	3F	D2	EC	C9	7A	AC	0D	CC	7F	D3	35	25	C2	6F
8	BF	86	07	91	5E	C5	75	4E	C1	83	E8	CA	B0	E9	75	B6
9	07	16	F8	7B	49	D5	FA	AD	5B	5F	C2	19	12	13	C5	20
A	99	B3	44	9F	F2	71	9B	38	7A	AE	A5	0D	48	C4	EF	1E
B	F4	09	8F	D1	2F	88	60	9B	E9	9F	75	66	69	7C	23	62
C	05	DE	30	DE	BB	D5	96	4D	52	AB	77	32	09	B4	3F	AB
D	FF	97	D6	FB	FE	59	DB	BA	15	1A	64	2D	02	F8	5D	3B
E	74	EE	1B	82	C8	63	F8	F4	BF	49	50	5A	7C	FC	46	47
F	C9	97	ED	52	59	D3	01	56	79	DB	C7	9A	E7	26	12	00

Şekil 2.4. S-Kutusu Örneği

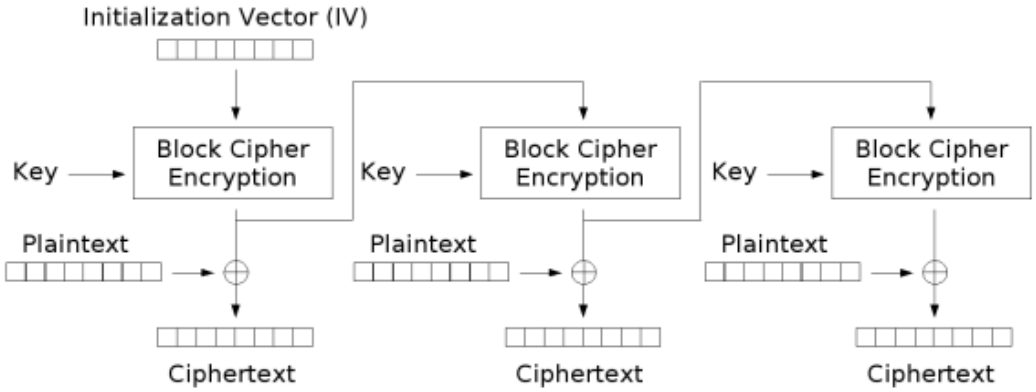
S-box'lar genellikle bir şifreleme algoritmasının anahtarı veya alt anahtarlarıyla birleştirilerek kullanılır. Bu, şifreleme işleminin her turunda farklı S-box'ların uygulanmasını sağlar, böylece şifreleme süreci daha karmaşık ve tahmin edilemez hale gelir.

Örnek olarak, DES (Data Encryption Standard) şifreleme algoritması, S-box'ları kullanarak veri bloklarını şifreler. DES'teki S-box'lar, 6-bit giriş değerlerini 4-bit çıkış

değerlerine dönüştürerek şifreleme işlemini karmaşıklaştırır. Bu, DES'in doğrusal kriptanaliz gibi saldırılara karşı dayanıklı olmasına yardımcı olur.

Initialization vector (IV), blok şifreleme algoritmalarında kullanılan, şifreleme ve şifre çözme işlemlerini birbirinden farklı hale getiren bir değerdir. IV, genellikle rastgele olarak oluşturulur ve her şifreleme işlemi için farklıdır.

IV, şifreleme işleminin ilk aşamasında, şifreleme anahtarı ile kullanılır. IV, şifreleme anahtarının her seferinde farklı bir şekilde kullanılmasını sağlar. Bu, kriptanalizi zorlaştırarak şifreleme algoritmasının güvenliğini artırır.



Şekil 2.5. Başlatma Vektörü

IV, blok şifreleme algoritmalarında kullanılan önemli bir güvenlik özelliğidir. Doğru tasarlanmış bir IV, kriptanalizi zorlaştırarak şifreleme algoritmasının güvenliğini artırabilir.

Genel Anahtar (Public key), asimetrik şifreleme algoritmalarında kullanılır. Asimetrik şifreleme algoritmaları, şifreleme ve şifre çözme işlemlerinde farklı anahtarlar kullanır. Bu, asimetrik şifreleme algoritmalarını, simetrik şifreleme algoritmalarına göre daha güvenli kılar.

Özel anahtar (Private key), şifreleme ve şifre çözme işlemlerinde kullanılan iki anahtardan biridir. Private key, yalnızca anahtarı sahibi tarafından bilinebilen ve paylaşılamayan bir anahtardır.

Özel anahtar, asimetrik şifreleme algoritmalarında kullanılır. Asimetrik şifreleme algoritmaları, şifreleme ve şifre çözme işlemlerinde farklı anahtarlar kullanır. Bu, asimetrik şifreleme algoritmalarını, simetrik şifreleme algoritmalarına göre daha güvenli kılar.

Tuzlama (Salting) şifreleme işlemine rastgele bir değer eklenerek, şifrelenmiş verinin daha güvenli hale getirilmesini sağlayan bir tekniktir. Salt, genellikle şifreleme algoritmasının kendisi tarafından üretilir.

Salting, şifreleme işlemini daha zor hale getirerek, brute force saldırılarına karşı koruma sağlar. Brute force saldırılarında, saldırgan, şifreyi oluşturan tüm olası değerleri deneyerek şifreyi kırmaya çalışır. Salting, olası değerlerin sayısını artırarak, brute force saldırılarını daha zor ve zaman alıcı hale getirir.

Blok şifreleme, kriptografide kullanılan bir şifreleme yöntemidir. Bu yöntemde, sabit uzunluktaki bit grupları üzerine simetrik anahtar ile belirlenmiş bir deterministik algoritmanın uygulanmasıdır. Blok şifreleme, büyük boyutlu verilerin şifrelemesinde yaygın biçimde kullanılmaktadır. Şifreleme işlemi, tekrarlanan bileşke şifre kullanılarak birden çok raund da gerçekleşir ve her raundda ana anahtar dan üretilmiş farklı bir alt anahtar kullanılır. Blok şifreleme, verileri bir seferde bir bit şifreleyen bir akış şifresinin aksine, sabit boyutlu blokları aynı anda işler (Wikiwand; Blok Şifreleme, 2020).

Blok şifreleme çeşitli modlara sahiptir. Bu modlar şifrelemede kullanılan anahataın bloklara nasıl uygulanacağına karar verir.

ECB (Electronic Codebook) modu, blok şifreleme algoritmalarını kullanarak veriyi şifrelemek için bir moddur. Bu mod, veriyi sabit boyuttaki bloklara böler ve her bloğu aynı anahtar ve şifreleme algoritması kullanarak bağımsız olarak şifreler.

ECB modu diğer modlara göre basit kabul edilir. ECB modu, her bloğun bağımsız olarak şifrelenmesinden dolayı paralel şifrelemeye izin verir. Bir bloğun şifrelenmesi esnasında hata oluşursa sadece o blok etkilenir.

Her blok birbirinden bağımsız olarak şifreleneceği için aynı değerlere sahip bloklar aynı şifre ile şifrelendiğinde aynı değerleri aynı değeri çıktı olarak verirler. Bunun sonucunda kriptoanalistler veri blokları arasında desen ortaya çıkartabilir.

CBC (Cipher Block Chaining) modu, blok şifreleme algoritmalarının çalışma modlarından biridir. Bu mod, her bloğun şifrelenmesinde önceki bloğun şifrelenmiş metni ile XOR işlemi yaparak çalışır. CBC modunsa şifreleme adımları şöyledir.

İlk olarak bir Initialization Vector (IV) belirlenir. IV, şifreleme işlemine başlarken kullanılan rastgele bir değerdir. Açık metin blokları, IV veya bir önceki bloğun

şifrelenmiş metni ile XOR işlemine tabi tutulur. XOR işleminden sonra elde edilen sonuç, şifreleme algoritması kullanılarak şifrelenir. Elde edilen şifrelenmiş blok, bir sonraki bloğun şifrelenmesinde kullanılmak üzere saklanır.

OFB (Output Feedback) modu, blok şifreleme algoritmalarında kullanılan bir şifreleme operasyon modudur. OFB modunda, şifreleme algoritması IV (İlk Değer) ve anahtar üzerinde işlem yapar ve ardından elde edilen çıktı, şifrelenecek veri ile XOR işlemine tabi tutulur. Bu işlem sonucunda elde edilen çıktı, şifreli veri olarak kullanılır. OFB modunun avantajlarından biri, keystream'in önceden üretilebilmesidir, çünkü plaintext'e bağlı değildir.

Cipher Feedback (CFB) modu, blok şifreleme algoritmalarında kullanılan bir çalışma modudur. CFB modunda, her blok, şifrelenen önceki bloktan elde edilen anahtar akışına XOR işlemine tabi tutularak şifrelenir. Bu, açık metindeki tekrar eden desenlerin şifreli metinde tekrar etmesini önler ve blok şifreleme algoritmasının güvenliğinden yararlanmayı sağlar.

CFB modunun çalışma prensibini şöyle özetleyebiliriz. İlk olarak, açık metin, blok şifreleme algoritmasının blok boyutuna bölünür. İlk blok, şifreleme anahtarı kullanılarak şifrelenir. Şifrelenmiş ilk blok, şifreleme anahtarı kullanılarak tekrar şifrelenir (Crypto 101 Şifreleme Operasyonu Modları – ECB, CBC, OFB – Mehmet INCE; Information Security, 2015). Şifrelenmiş ikinci blok, şifrelenmiş ilk bloktan elde edilen anahtar akışına XOR işlemine tabi tutularak şifrelenir. Bu işlem, açık metindeki son bloka kadar tekrarlanır. Şifrelenmiş bloklar birleştirilerek şifreli metin oluşturulur.

Açık metinde tekrar eden desenlerin şifreli metinde tekrar etmemesi ve CBC moduna göre daha hızlı olması CFB modunun avantajlarındadır.

Counter (CTR) modu, blok şifreleme algoritmalarında kullanılan bir çalışma modudur. CTR modunda, her blok, bir sayaçla XOR işlemine tabi tutularak şifrelenir. Bu, açık metindeki tekrar eden desenlerin şifreli metinde tekrar etmesini önler ve blok şifreleme algoritmasının güvenliğinden yararlanmayı sağlar.

CTR modunun aşamaları şöyle özetlenebilir. Açık metin, blok şifreleme algoritmasının blok boyutuna bölünür. Bir sayaç oluşturulur. İlk blok, şifreleme anahtarı kullanılarak şifrelenir. Şifrelenmiş ilk blok, sayaçla XOR işlemine tabi

tutularak şifreli metin oluşturulur. Bu işlem, açık metindeki son bloka kadar tekrarlanır. Şifreli bloklar birleştirilerek şifreli metin oluşturulur.

CTR modunun CBC ve CFB modlarından farkı, her blokta şifrelenen önceki bloktan elde edilen anahtar akışının değil, bir sayacın kullanılmasıdır. Bu, CTR modunu CBC ve CFB modlarına göre daha hızlı yapar. Ayrıca, CTR modu, blok şifreleme algoritmasının blok boyutundan bağımsızdır.

Kriptografide, Galois/Counter Mode (GCM), blok şifreleme algoritmalarında kullanılan bir çalışma modudur. GCM, hem veri gizliliği (gizlilik) hem de veri bütünlüğü (bütünlük) sağlamak için tasarlanmış kimliği doğrulanmış bir şifreleme algoritmasıdır.

GCM modunun aşamaları şöyledir. Açık metin, blok şifreleme algoritmasının blok boyutuna bölünür. Bir sayaç oluşturulur. İlk blok, şifreleme anahtarı kullanılarak şifrelenir. Şifrelenmiş ilk blok, sayaçla XOR işlemine tabi tutularak şifreli metin oluşturulur. Bu işlem, açık metindeki son bloka kadar tekrarlanır. Şifreli bloklar birleştirilerek şifreli metin oluşturulur. Tag, şifrelenmiş metin ve şifreleme anahtarı kullanılarak hesaplanır.

GCM modunun CBC, CTR ve CFB modlarından farkı, tag'in kullanılmasıdır. Tag, şifrelenmiş verinin bütünlüğünü ve doğruluğunu doğrulamak için kullanılır. GCM modu, tag'i kullanarak, şifrelenmiş verinin değiştirilmesi veya bozulması durumunda, şifrelenmiş verinin doğruluğunu doğrulayabilir (Wikimedia projelerine katkıda bulunanlar, 2020).

Akış şifrelemesi, açık metni tek bit veya bayt halinde şifreleyen simetrik anahtarlı bir şifreleme algoritmasıdır. Blok şifrelemesi gibi açık metni sabit boyutlu bloklar halinde şifreleyen bir blok şifrelemesinden farklı olarak, akış şifrelemesi açık metni sürekli bir akış halinde şifreler. Bu, akış şifrelerini düşük gecikmelerin önemli olduğu uygulamalar için uygun hale getirir, örneğin gerçek zamanlı iletişim ve akışlı multimedya (What Is Stream Cipher?; Definition, Attacks, and More, 2020).

Akış şifreleme algoritmasının da her bit ayrı şifrelendiği için düşük gecikmeye sahiptir. Açık metin sabit bir blok boyutuna sığdırılmak zorunda olmadığı için blok şifreleme algoritmalarına göre daha verimlidir.

Authentication Tag (Doğrulama Etiketi), şifreleme işlemlerinde kullanılan bir mekanizmadır ve özellikle authenticated encryption (doğrulanmış şifreleme) modları

ile birlikte kullanılır. Bu etiket, şifreleme işlemi sonrasında verinin bütünlüğünü doğrulamak ve kimlik doğrulama sağlamak amacıyla kullanılır. Doğrulama etiketi, genellikle kullanılan şifreleme algoritmalarının modlarına bağlı olarak farklı isimlerle anılabilir, örneğin, MAC (Message Authentication Code) veya HMAC (Hash-based Message Authentication Code).

Doğrulama etiketinin üretilme ve tüketilme süreci şöyledir. İlk olarak, orijinal veri belirli bir şifreleme algoritması ve anahtar kullanılarak şifrelenir. Bu şifreleme işlemi, mesajın gizliliğini sağlamak amacıyla gerçekleştirilir What Is the Purpose of an Authentication Tag in AEAD Encryption Schemes?, 2020). Doğrulama etiketi, şifreleme işleminden çıkan şifrelenmiş veriye dayanarak oluşturulur. Şifrelenmiş verinin üzerine bir doğrulama kodu eklenir. Bu doğrulama kodu, verinin bütünlüğünü ve kimliğini doğrulamak için kullanılır. Authentication Tag, şifrelenmiş veri ile birlikte iletilen bir bileşen olarak eklenir. Şifrelenmiş veri ve Doğrulama etiketi birlikte alıcıya iletilir. Alıcı, şifrelenmiş veriyi alır ve kendi anahtarını kullanarak şifreleme işleminden geçirir. Ardından, Doğrulama etiketini kullanarak verinin bütünlüğünü ve kimliğini doğrular. Eğer Doğrulama etiketi doğrulanamazsa, alıcı şifrelenmiş veriyi güvenli bir şekilde çözemez.

Doğrulama etiketi, şifreleme işleminin sadece gizliliği değil, aynı zamanda bütünlüğü ve kimliği de sağlamasını amaçlar. Bu, verinin şifrelenmiş olduğu ve aynı zamanda değiştirilmediği ve doğru kaynaktan geldiği konusunda güvence sağlar. Doğrulama etiketi, güvenli iletişim ve veri güvenliği uygulamalarında sıkça kullanılır (Editor, 2020).

Şifrelemede ek veri (Additional Data veya AAD), şifreleme işlemine katılan, ancak şifrelenmeyen veri parçasını ifade eder. Ek veri, özellikle authenticated encryption (doğrulanmış şifreleme) modları veya şifreleme ile kimlik doğrulama yöntemleri kullanılırken önemli bir kavramdır.

Şifrelemede (dolgulama) padding, şifrelenen verinin uzunluğunu sabit tutmak için kullanılan bir tekniktir. Şifreleme algoritmaları genellikle belirli bir blok boyutunda çalışır. Bu nedenle, şifrelenen verinin uzunluğu bu blok boyutuna eşit veya katıdır.

Padding, şifrelenen verinin uzunluğunu bu blok boyutuna eşit veya kat yapacak şekilde ek bilgi ekleme işlemidir. Bu ek bilgi genellikle rastgele veya tahmin edilmesi zor olan bir değerdir.

Padding, şifreleme işleminin daha verimli olmasını sağlar. Şifreleme algoritmaları, şifrelenecek verinin uzunluğuna göre daha az veya daha fazla işlem gerçekleştirebilir. Padding, şifrelenen verinin uzunluğunu sabit tutarak, şifreleme algoritmasının her zaman aynı işlem sayısını gerçekleştirmesini sağlar.

Padding, şifreleme işleminin güvenliğini de artırabilir. Şifrelenen verinin uzunluğu sabit tutulursa, şifre çözme işlemini gerçekleştiren saldırgan, şifrelenen verinin uzunluğunu kullanarak şifrelenen verinin bir kısmını tahmin etmeye çalışabilir. Padding, bu saldırıyı önlemeye yardımcı olabilir.

Şifrelemede kullanılan padding teknikleri şunlardır:

- **Null padding**

Bu teknikte, şifrelenen verinin uzunluğunu sabit tutmak için verinin sonuna rastgele veya tahmin edilmesi zor olan bir değer eklenir.

- **Zero padding**

Bu teknikte, şifrelenen verinin uzunluğunu sabit tutmak için verinin sonuna sıfır eklenir.

- **PKCS#5 padding**

Bu teknikte, şifrelenen verinin uzunluğunu sabit tutmak için verinin sonuna rastgele veya tahmin edilmesi zor olan bir değer eklenir. Bu değer, verinin uzunluğunu blok boyutuna eşit veya kat edecek şekilde seçilir.

PKCS#5 padding, şifrelemede en yaygın kullanılan padding tekniğidir.

Padding, şifreleme işleminin verimliliğini ve güvenliğini artıran önemli bir tekniktir.

2.4.3. Geleneksel şifreleme algoritmaları

Geleneksel şifreleme yöntemleri ile açık veri üzerinde çeşitli matematiksel işlemler gerçekleştirerek veri bütünlüğü korunarak veri gizliliği sağlanır. Şifreleme algoritmalarının temel özellikleri veri gizliliği sağlama, veri bütünlüğünü sağlama ve gizlenen verinin sadece yetkisi olanlar tarafından erişilmesini sağlamaktır.

AES, Advanced Encryption Standard'ın (Gelişmiş Şifreleme Standardı) kısaltmasıdır. 2001 yılında ABD Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından geliştirilmiş ve 2002 yılında güvenli bir blok şifreleme algoritması olarak

standartlaştırılmıştır. AES, günümüzde en yaygın kullanılan blok şifreleme algoritmalarından biridir.

AES, 128, 192 veya 256 bit uzunluğunda anahtarlar kullanarak çalışır. Anahtar uzunluğu, şifreleme işleminin güvenliğini belirler. Daha uzun anahtar, daha güvenli şifreleme sağlar. AES, 128 bitlik bir blok boyutunda çalışır. Bu, şifrelenecek verinin 128 bitlik bloklar halinde şifrelendiği anlamına gelir (Hussien, 2021).

2.4.4. Hafif sıklet şifreleme algoritmaları

Hafif şifreleme algoritmaları, özellikle IoT (Nesnelerin İnterneti) cihazları gibi kaynakları sınırlı olan cihazlarda kullanılmak üzere tasarlanmıştır. Bu algoritmalar, verileri hızlı ve güvenli bir şekilde şifrelemek için tasarlanmıştır. Hafif şifreleme algoritmaları, AES (Advanced Encryption Standard) gibi diğer şifreleme algoritmalarına göre daha az kaynak tüketir ve daha hızlıdır. (Sadhukhan ve ark, 2017) Bu algoritmaların güvenli bir şekilde kullanılabilmesi için yeterli güvenlik seviyelerine sahip olmaları gerekmektedir. Hafif şifreleme algoritmaları arasında, PRESENT, CHACHA, SPECK, ASCON gibi birçok algoritma bulunmaktadır. Çalışmamızda Ascon, Speck ve Chacha hafif sıklet şifreleme algoritmaları ele alınmıştır.

Speck, hafif sıklet bir blok şifreleme algoritmasıdır. Sınırlı kaynaklara sahip cihazlar için tasarlanmış olan bu algoritma, özellikle Internet of Things (IoT) cihazları gibi düşük güç tüketimine ve sınırlı işlem kapasitesine sahip uygulamalar için uygundur. Speck algoritması, NSA (National Security Agency) tarafından geliştirilmiştir ve 2013 yılında kamuoyuna açıklanmıştır (Beaulieu ve ark, 2015).

Ascon, NIST tarafından yürütülen Lightweight Cryptography Standardization (LWC) yarışmasının finalistlerinden biridir. Ascon, güçlü güvenlik, hafiflik ve performans özelliklerine sahip bir şifreleme algoritmasıdır.

Ascon, 128, 192 ve 256 bitlik anahtar boyutları ile kullanılabilir. Bu anahtar boyutları, gelişmiş bir kriptoanalize karşı dayanıklı olmasını sağlar. Ascon, kriptoanalizde kullanılan birçok saldırıya karşı dayanıklı olduğu kanıtlanmıştır.

Ascon, özellikle düşük güç ve bellek kaynaklarına sahip cihazlarda kullanım için tasarlanmıştır. Ascon, yalnızca 128 bitlik bir anahtar için 1440 işlem gerektirir. Bu, AES gibi diğer yaygın şifreleme algoritmalarından daha az güç ve bellek kullanır.

Ascon, yüksek performansa sahip bir şifreleme algoritmasıdır. Ascon, AES'e benzer bir performansa sahiptir. Ascon, 128 bitlik bir anahtar için saniyede 3000 bit şifreleme yapabilir (Dobraunig ve ark, 2021).

Chachapoly şifreleme, Chacha20 ve Poly1305 şifreleme algoritmalarının birleşiminden oluşan bir şifreleme algoritmasıdır. Chacha20, bir gerçek zamanlı şifreleme algoritmasıdır. Poly1305 ise bir mac algoritmasıdır.

Chacha20, 256 bitlik bir anahtar ve 256 bitlik bir IV (initialization vector) ile çalışır. Chacha20, 64 bitlik bloklar halinde çalışır ve her blok için 20 tur kullanır. Chacha20, güçlü güvenlik ve yüksek performans özelliklerine sahiptir.

Poly1305, 256 bitlik bir anahtar ve 128 bitlik bir mesaj ile çalışır. Poly1305, 64 bitlik bloklar halinde çalışır ve her blok için 10 tur kullanır. Poly1305, güçlü güvenlik ve yüksek performans özelliklerine sahiptir.

Chachapoly algoritması, ilk olarak Chacha20 algoritmasını kullanarak mesajı şifreler. Chacha20 algoritması, mesajı şifrelemek için bir dizi tur kullanır. Her tur, bir dizi karıştırma ve XOR işlemi içerir.

Chachapoly algoritması, daha sonra Poly1305 algoritmasını kullanarak şifrelenmiş mesajın bir mac'ini hesaplar. Poly1305 algoritması, şifrelenmiş mesajın bir mac'ini hesaplamak için bir dizi tur kullanır. Her tur, bir dizi işlem içerir.

Chachapoly algoritması, şifrelenmiş mesaj ve mac'i birlikte gönderir. Alıcı, önce şifrelenmiş mesajı Chacha20 algoritmasını kullanarak çözer. Daha sonra, çözülmüş mesajın mac'ini Poly1305 algoritmasını kullanarak hesaplar. Eğer iki mac aynıysa, mesajın doğru bir şekilde şifrelendiği ve çözüldüğü anlaşılır.

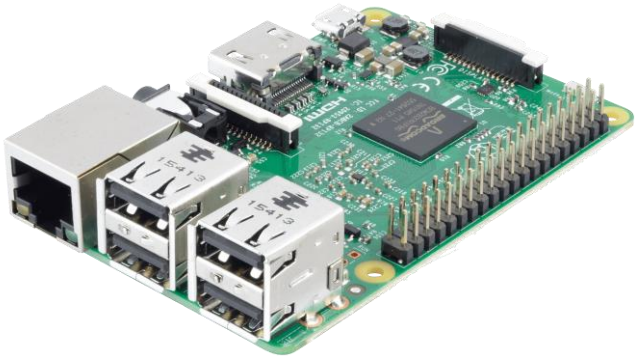
3. MATERYAL VE YÖNTEM

3.1. Donanımlar

Çalışmamızda akıllı ev simülasyonunu gerçekleştirmek için çeşitli IoT cihazları kullanılmıştır. Akıllı evlerdeki kontrol merkezini modellemek için Raspberry Pi3 cihazı kullanılmıştır. Akıllı evlerdeki akıllı ev cihazlarını (akıllı termostat, akıllı çamaşır makinesi, akıllı buzdolabı gibi) modellemek için NodeMCU cihazı kullanılmıştır.

Raspberry Pi 3, Raspberry Pi Vakfı tarafından geliştirilen, düşük maliyetli, tek kartlı bir bilgisayar platformudur. Raspberry Pi serisi, özellikle öğrenme, prototipleme, hobi projeleri ve düşük güç tüketimine sahip uygulamalar için popüler olan bir dizi küçük bilgisayar kartından oluşur.

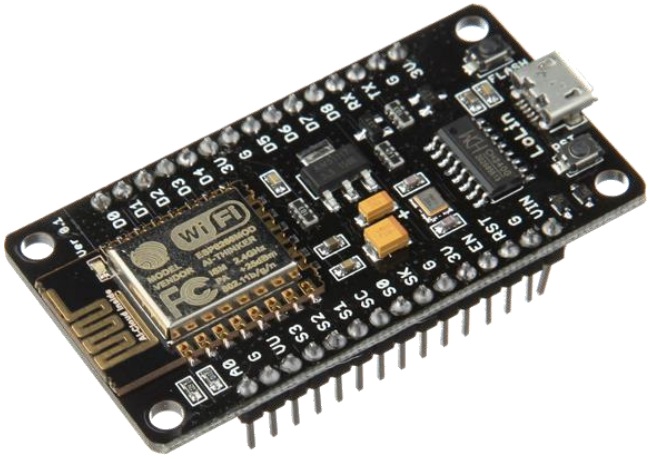
Raspberry Pi 3, 1.2 GHz dört çekirdekli ARM Cortex-A53 işlemciye sahiptir. 1 GB LPDDR2 SDRAM' e sahiptir. Raspberry Pi 3, dahili olarak Wi-Fi (802.11n) ve Bluetooth 4.1 özelliklerine sahiptir. Bu, kablosuz bağlantılar ve Bluetooth cihazlarla iletişim kurma yeteneği sağlar. HDMI çıkışı: Full HD (1080p) video çıkışı desteklenir. USB portları: Raspberry Pi 3, dört adet USB 2.0 portuna sahiptir. Ethernet bağlantısı: 10/100 Mbps Ethernet bağlantısı bulunmaktadır. GPIO (Genel Amaçlı Giriş/Çıkış) pinleri: 40 adet GPIO pini, çeşitli projelerde kullanılabilecek genişleme ve bağlantı noktaları sunar. MicroSD kart üzerinden çalışır. İşletim sistemi ve kullanıcı verileri genellikle bir MicroSD kart üzerine yüklenir.



Şekil 3.1. Raspberry Pi3 Cihazı

NodeMCU, Espressif Systems tarafından geliştirilen bir açık kaynaklı, Lua tabanlı ürün yazılımı ve geliştirme kartıdır. NodeMCU, özellikle IoT tabanlı uygulamalar için tasarlanmıştır.

MCU, "Mikrodenetleyici"nin kısaltmasıdır. Mikrodenetleyiciler, belirli görevleri yapmak üzere tasarlanmış entegre devrelerdir. Genellikle I/O (Input/Output) pinleri bulunur ve elektronik cihazlarda, otomotiv sistemlerinde, endüstriyel kontrol uygulamalarında ve diğer birçok alanda kullanılırlar. Mikrodenetleyiciler, karmaşık lojik fonksiyonların toplanarak entegre edildiği bir cihazdır. Mikrodenetleyiciler, genellikle başlı başına bir sistem olarak kullanılırlar ve CPU (Central Processing Unit) ile çevre birimlerini içerirler.



Şekil 3.2. NodeMCU Cihazı

Nodemcu'nun temel özellikleri şu şekildedir. NodeMCU, 11 adet I/O pinine sahiptir.

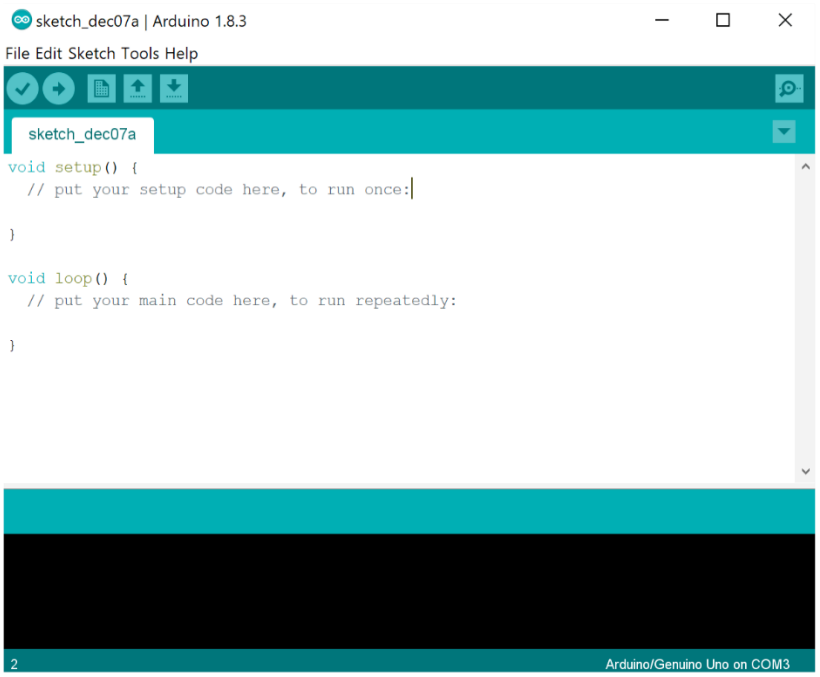
1 adet analog ve 1 adet ADC kanalı bulunur. NodeMCU, UART, SPI ve I2C gibi iletişim protokollerini destekler. Dahili olarak 802.11 b/g/n Wi-Fi özelliği bulunur. NodeMCU, 3.3V çalışma voltajına sahiptir. Ayrıca, 4 MB flash bellek ve 64 KB SRAM bulunur. NodeMCU, -40C ila 125C arasında çalışma sıcaklığına sahiptir (admin, 2022).

3.2. Yazılımlar

Akıllı ev modelleme sisteminin oluşturulma aşamasında çeşitli yazılım uygulamalarından faydalanılmıştır.

Arduino IDE (Integrated Development Environment- Entegre Geliştirme Ortamı), Arduino mikrodenetleyicileri için bir ücretsiz, açık kaynaklı, platformlar arası entegre

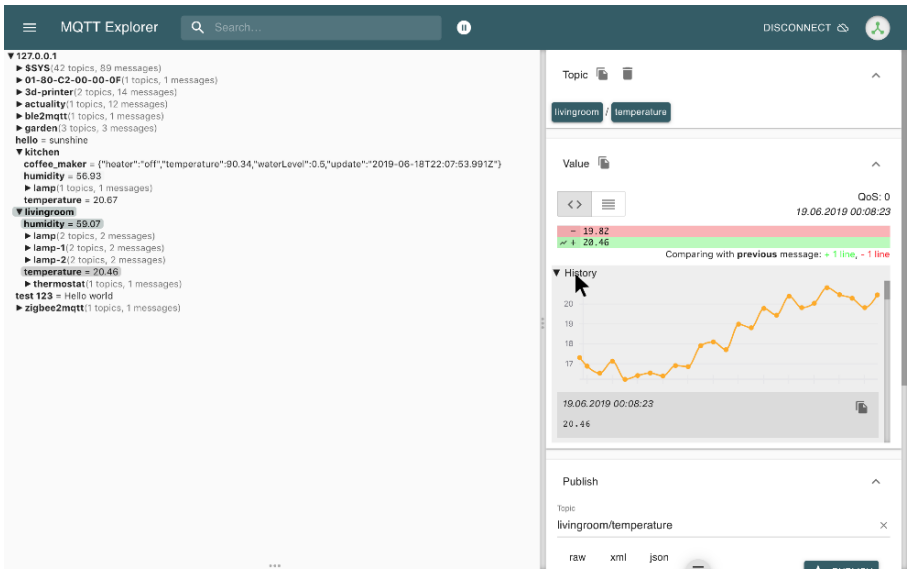
geliştirme ortamıdır. C ve C++ dillerini destekler. Arduino IDE, Arduino mikrodenetleyicileri için kod yazmak, derlemek ve yüklemek için kullanılır. Arduino kütüphanelerinin kolay kullanımına imkân sağlar.



Şekil 3.3. Arduino Programı

MQTT Explorer, MQTT (Message Queuing Telemetry Transport) protokolü üzerinden çalışan IoT (Internet of Things) cihazları ve MQTT sunucularıyla etkileşimde bulunmak için kullanılan bir masaüstü uygulamasıdır. MQTT, hafif, güvenilir ve düşük bant genişliği tüketen bir mesajlaşma protokolüdür ve genellikle IoT cihazları arasında veri iletimi için tercih edilir. MQTT Explorer, MQTT tabanlı sistemlerin izlenmesi, keşfedilmesi ve test edilmesi için bir araç sağlar.

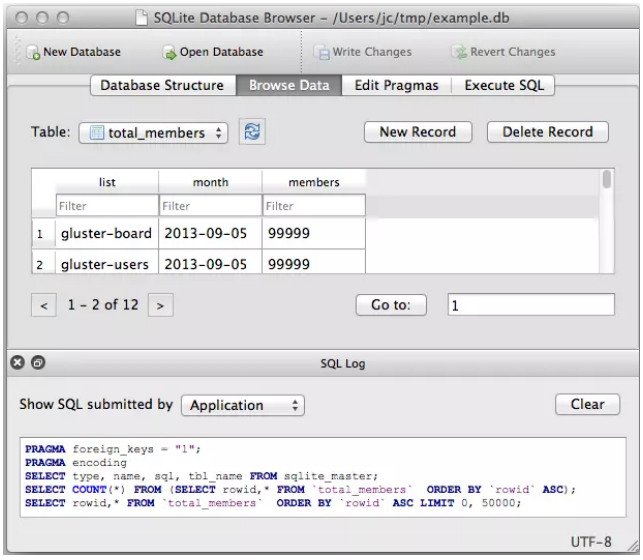
MQTT Explorer özelikleri şöyledir. MQTT brokerlarını ve ilgili konuları izleyebilir. Bu, cihazların ve sunucuların durumunu takip etmek ve veri alışverişini gözlemlemek için kullanışlıdır. Kullanıcılar, belirli bir MQTT konusuna abone olabilir veya bir konuya veri yayınlayabilirler. Bu, sistemdeki cihazlarla etkileşimde bulunmayı sağlar (Nordquist, 2020).



Şekil 3.4. MQTT Explorer Programı

SQLite, dünyada en çok dağıtılan ve tavsiye edilen kaynak kodları halka açık, sunucu yazılımı ve yapılandırma gereksinimi olmayan ilişkisel bir SQL veri tabanı motorudur. SQLite, C/C++ programlama dilleriyle geliştirilmiştir ve birçok programlama dili içerisinde rahatlıkla kullanılabilir. SQLite, sorgulama ve depolama özelliklerini, dosya sistemi erişiminin kullanım kolaylığı ile birleştirmek istendiğinde en iyi sonucu verir. SQLite veri tabanları taşınabilir ve yedeklenmesi kolaydır. Sunucu bağlantısı olmadığından, bir geliştirme ortamında olmasak bile SQLite'ı yapılandırmayı kolay bir hale getirir (DB Browser for SQLite, 2019).

SQLite DB Browser, SQLite veritabanı dosyalarını oluşturmak, tasarlamak ve düzenlemek için kullanılan bir araçtır 1. Bu araç, kullanıcıların ve geliştiricilerin veritabanlarını oluşturmasına, aramasına ve düzenlemesine olanak tanır. SQLite DB Browser, kullanımı kolay bir arayüz sunar. SQLite DB Browser, açık kaynak kodlu bir araçtır ve Windows, Mac ve Linux işletim sistemleri için kullanılabilir (Mills, 2022).



Şekil 3.5. SQLite Programı

VNC, Virtual Network Computing'in kısaltmasıdır. Bir ağ üzerinden uzak bir bilgisayarın masaüstünü kontrol etmek için kullanılan bir yazılım protokolüdür. VNC, grafik arabirimli uygulamaları çalıştırmanıza, dosyaları aktarmanıza ve uzak bilgisayarı yönetmenize olanak tanır.

VNC, platformlar (Windows, macOS, Linux gibi) arası kullanımı destekler. Şifreli bağlantı özelliği sayesinde güvenli veri iletimi sağlar.

Raspbian OS, Raspberry Pi için optimize edilmiş Debian'a dayalı özgür bir işletim sistemidir. Raspbian, Unix/Linux işletim sistemi ailesine aittir ve paket yöneticisi olarak dpkg kullanır. Son kararlı sürümü Raspberry Pi OS Bullseye'dir. Raspberry Pi OS, PIXEL adlı kullanıcı arayüzüne sahiptir. Ayrıca, Raspberry Pi OS, ARM ve i386 platformlarını destekler.

3.3. Protokoller

3.3.1. MQTT

MQTT, makineden makineye iletişim için kullanılan, standart tabanlı bir mesajlaşma protokolüdür. IoT cihazları arasında veri iletimini kolaylaştırmak için kullanılır. MQTT, düşük bant genişliği ve düşük gecikme süresi gibi özelliklere odaklanarak, düşük güç tüketimi gerektiren cihazlar arasında veri iletişimini etkili bir şekilde yönetir (KAZANCI, 2021).

MQTT özellikleri şöyledir. Yayıncı-abone (publish-subscribe) mimarisi kullanır. Bu sayede birden fazla istemcinin aynı anda bağlı olmasını gerektirmez ve mesajları

abonelere iletebilir. Konu dizeleri, özel bir sınırlayıcı karakter olan eğik çizgi (/) kullanılarak doğal bir konu ağacı oluşturur. Bu sayede istemciler, özel joker karakterleri kullanarak konu ağacındaki tüm dallara abone olabilir ve abonelikten çıkabilir. İstemcilere üç hizmet seviyesi arasında seçim yapma olanağı sağlayan QoS özelliğine sahiptir. Bu özellik, mesajların göndereni ile alıcısı arasındaki teslimat garantisini tanımlar (MQTT Nedir ve Nasıl Çalışır?, 2020).

Mesajlar yayınlayan cihazlar yayıncılar olarak adlandırılır ve mesajları alan cihazlar aboneler olarak adlandırılır.

Bir yayıncı, bir mesajı konu adı ile yayınlar. Konu adı, mesajın ne hakkında olduğunu gösterir. Aboneler, ilgilendikleri konulara abone olabilirler. Bir abone, abone olduğu bir konu adına bir mesaj yayınlandığında, bu mesajı alır.

MQTT, yayıncılar ve aboneler arasında veri alışverişi için basit ve etkili bir yol sağlar. Bu mimari, IoT cihazları arasında veri alışverişi için yaygın olarak kullanılır.

Örneğin, bir sıcaklık sensörü, sıcaklık verilerini bir MQTT broker'a yayınlayabilir. Bu verilere abone olan bir akıllı ev uygulaması, sıcaklık verilerini kullanarak kullanıcıya bir uyarı gönderebilir.

3.3.2. HTTP

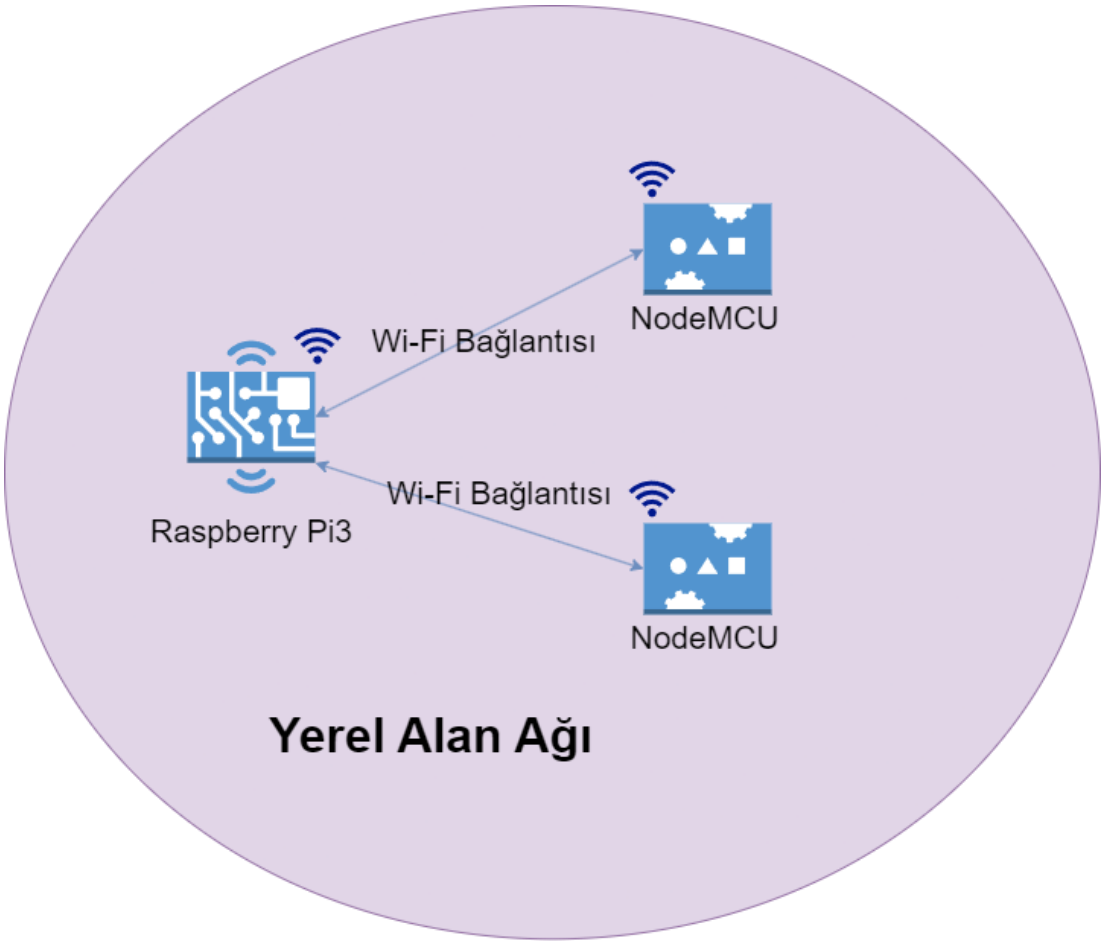
HTTP, Hypertext Transfer Protocol'ün kısaltmasıdır. İnternet üzerinden metin, resim, video, ses ve diğer verileri aktarmak için kullanılan bir iletişim protokolüdür. HTTP, web sitelerini görüntülemek, e-posta göndermek ve dosya indirmek gibi günlük olarak yaptığımız birçok şey için gereklidir.

HTTP, istemci-sunucu mimarisi kullanır. Bu, bir istemcinin bir sunucudan bilgi istemesine ve sunucunun istemciye bilgi göndermesine olanak tanır. HTTP istemcileri, bir sunucudan bilgi istemek için çeşitli istemci istekleri kullanabilir. Bu istemci isteklerini arasında GET, POST, PUT ve DELETE bulunur. HTTP sunucuları, istemci isteklerine çeşitli sunucu yanıtları ile yanıt verebilir. Bu sunucu yanıtları arasında 200 OK, 404 Not Found ve 500 Internal Server Error bulunur (HTTP Nedir? Ne İşe Yarar? HTTPS Ile Farkı Nedir? ;Pomelo Soft - Pomelo Soft, 2020).

HTTP, HTTPS'nin güvensiz bir versiyonudur 3. HTTPS, HTTP'nin güvenli bir versiyonudur ve bu protokolü kullanan bir web sitesinde gönderilen veriler şifrelenir ve doğrulanır.

3.4. Yöntem

Çalışmamızda örnek bir akıllı ev sistemi simülasyonu gerçekleştirilmiştir. Simülasyonda, evlerde bulunan cihazlar simüle edilmiştir. Akıllı ev cihazlarının akıllı ev kontrol birimiyle iletişim halinde olduğu bir modelleme gerçekleşmiştir. Cihazlar arasında sağlanan haberleşme esnasında hafif sıklet şifreleme algoritmaları kullanılmıştır. Şifreleme işleminde kullanılan algoritmalar Ascon, Aes, Speck ve Chacha'dir. Çalışmamızın sonucu olarak kullanılan şifreleme algoritmalarının performans analizi elde edilmiştir.

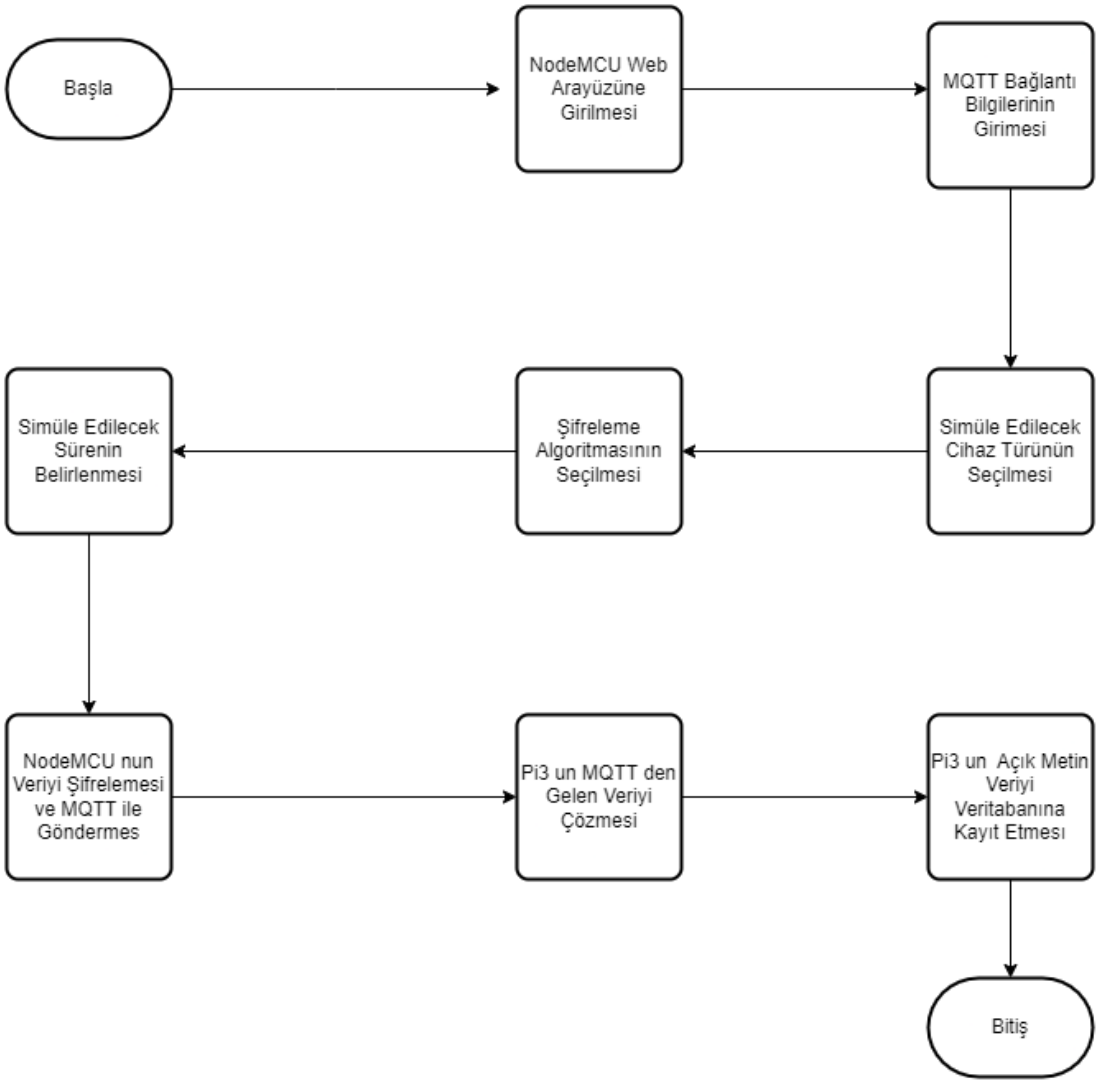


Şekil 3.6. Simülasyon Ağ Mimarisi

Şekilde görüldüğü gibi modellemede iki adet NodeMCU ve bir adet Raspberry Pi3 cihazı vardır. Cihazlar ev ekosistemi içinde kablosuz olarak haberleşmektedir. NodeMCU cihazları, günümüzde evlerde kullanılan akıllı ev cihazlarının simüle etmektedir. Raspberry Pi3 cihazı ise akıllı kontrol panelini simüle etmektedir.

Nodemcu cihazı çeşitli görevleri yerine getirmektedir. Simülasyon aşamasında HMI olarak bir web arayüzüne sahiptir. Web server'ı olarak gelen isteklere ilgili web sayfayı göndermektedir. Cihazın Flash hafızasında MQTT bilgilerini ve cihazında

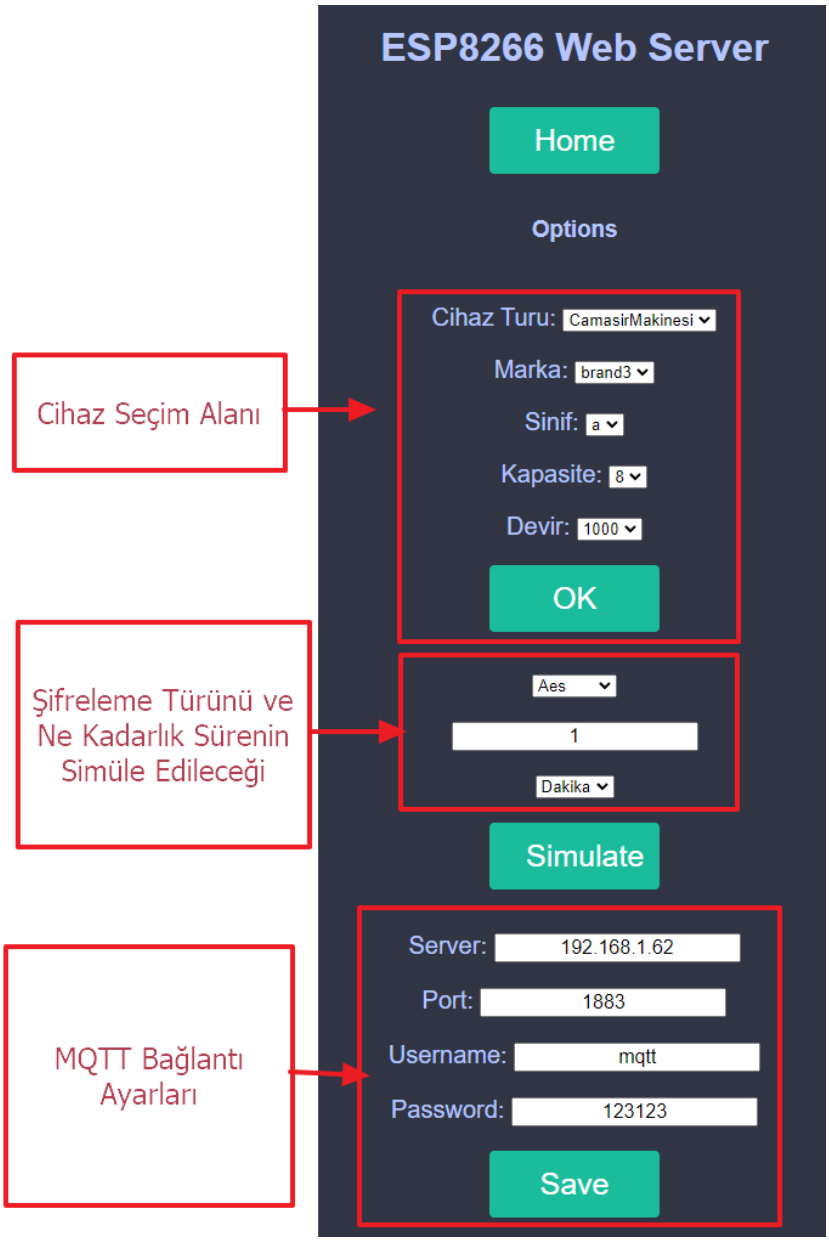
temsil edeceği akıllı ev cihazının bilgilerini saklamaktadır. Temsili akıllı ev cihazı bilgileri Json formatında tutulmaktadır. Verinin içeriğinde temsil edilecek cihazın türü, markası, enerji sınıfı, cihazın türüne özgü bilgiler ve dakikadaki kilowatt değeri yer almaktadır.



Şekil 3.7. Simülasyon Akış Diyagramı

Şekildeki akış diyagramında görüldüğü gibi simülasyonun gerçekleşme aşamaları şöyledir. NodeMCU ve Raspberry Pi3 cihazları aynı kablosuz ağa bağlantısı sağlanmıştır. NodeMCU cihazının sağlamış olduğu web arayüzüne tarayıcı ile erişilmiştir. Raspberry Pi3 cihazı üzerinde hizmet vermekte olan mqtt servise erişim sağlanabilmesi için gereken bilgiler arayüzden girilmiştir. MQTT yapılandırması için gerekli olan bilgiler Pi3 cihazının yerel ağdaki IP adresi, MQTT servisinin Pi3 cihazında çalıştığı port numarası ve MQTT servisine erişim için gerekli olan kullanıcı adı ve şifre bilgileridir. Arayüz üzerinden nodeMCU cihazının temsil edeceği cihazın

özellikleri seçilmiştir. Seçilen özelliklere göre cihazın enerji tüketim değerleri farklılık göstermektedir. Uygulamasının yapmış olduğumuz hafif sıklet şifreleme algoritmalarından Ascon, Speck, Chacha ve geleneksel şifreleme algoritmalarından Aes arasından seçim yapılmıştır. Temsili cihazında ne kadarlık süre çalıştığını belirtmek için süre bilgisi girilmiştir. Temsili cihaz verilerinde dakikada kaç kilowatt enerji tükettiği veri bulunmaktadır. Girilen süre bilgisine göre nodeMCU cihazı ilgili hesaplamayı yapar. Ve ardında Json formatında çıktı verisini üretir. Ardında seçilen şifreleme algoritmasına göre şifreleme işlemin gerçekleştirir. MQTT protokolü ile şifrelenmiş veriyi Pi3 cihazına gönderir. Pi3 cihazı MQTT servisinden gelen şifreli veriyi çözer. Ardında açık metni SQLite veri tabanın kaydeder.

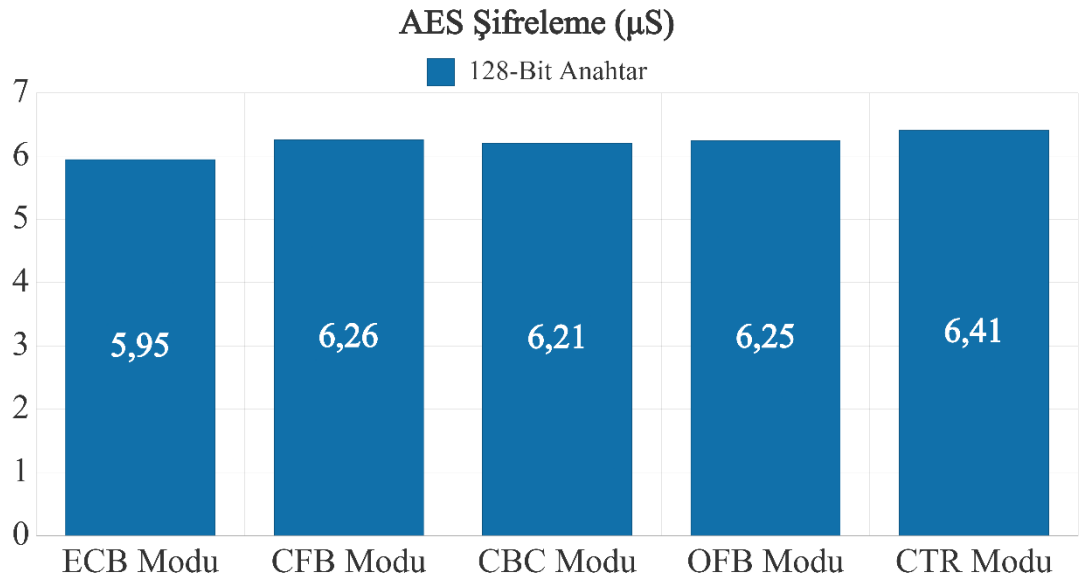


Şekil 3.8. NodeMCU Web Arayüzü

Şekilde web arayüzünde simülasyon esnasında kullanılan alanlar belirtilmiştir. Cihaz seçim alanında beş adet açılan menü vardır. Bu menüler kullanılarak NodeMCU'nun temsil edeceği akıllı ev aleti seçilir. Şifreleme türü alanında Aes, Speck, Chacha ve Ascon algoritmaları arasında seçim yapılır. Algoritma seçiminden sonra simülasyon süresi girilir. MQTT bağlantı alanında Pi cihazına ait olan MQTT bağlantı bilgileri girilir.

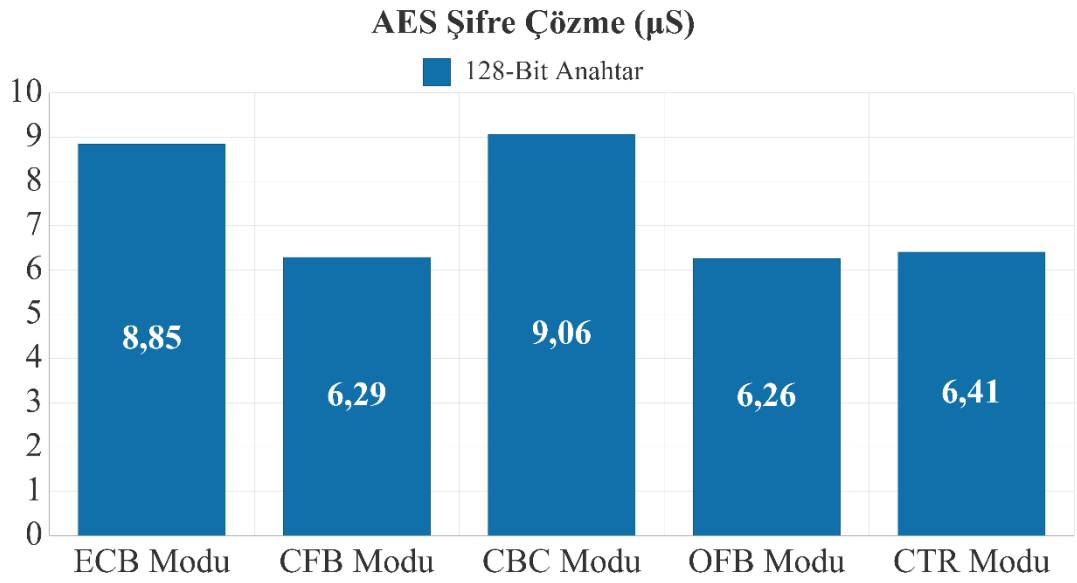
3.5. Bulgular

NodeMCU cihazları kullanarak şifreleme gerçekleştirildi. Gerçekleştirilen şifreleme sonucunda zamana bağlı performans verileri elde edildi. Gerçekleştirilen şifreleme ve şifre çözme işlemleri sonucunda hangi şifreleme algoritmasının daha performanslı olduğunun belirlenmesi amaçlanmıştır. Geleneksel şifreleme algoritmalarından Aes kullanılmıştır. Hafif sıklet şifreleme algoritmalarında Ascon, Chacha ve Speck algoritmaları kullanılmıştır. Kullanılan şifreleme algoritmaları simetrik şifreleme algoritmalarıdır. Şifreleme algoritmaları 128 bit, 192 bit ve 256 bit anahtar uzunluğunu desteklemektedir. Aes ve Speck şifreleme algoritmaları blok şifreleme algoritmalarıdır. Chacha ve Asco şifreleme algoritmaları akış şifreleme algoritmalarıdır. Blok şifreleme yöntemlerinde blok boyutuna sığmayan verileri şifreleyebilmek için çeşitli modlar uygulanır. ECB modu, CFB modu, CBC modu, OFB modu ve CTR modu Aes ve Speck şifreleme algoritmaları üzerinde uygulanan modlardır.



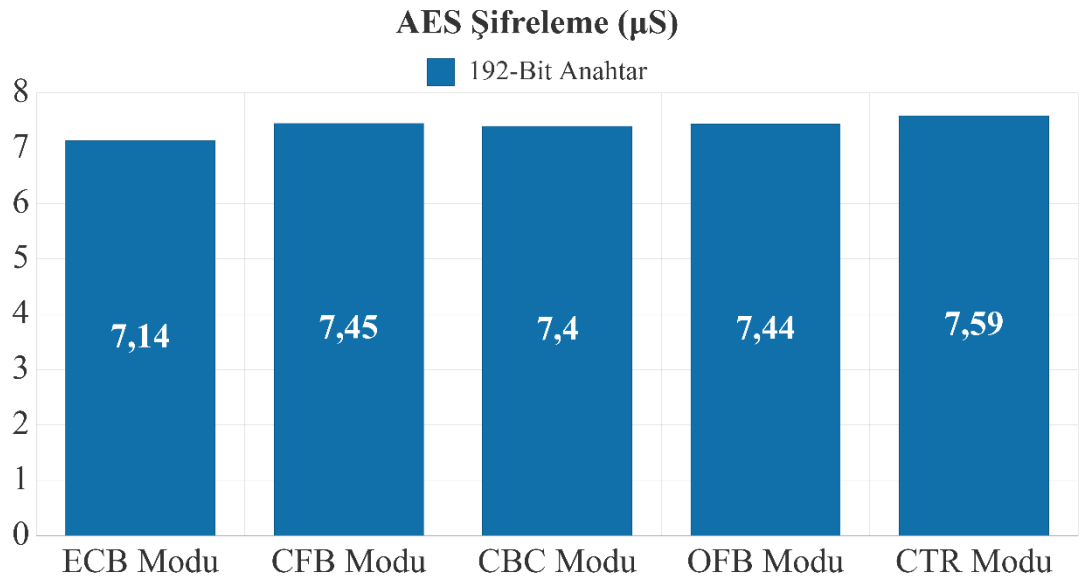
Şekil 3.9. Aes 128 Bit Anahtar Şifreleme

Aes algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifreleme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir. Aes algoritması 128 bit anahtar kullanarak uygulanmıştır. Aes algoritması bir byte'lık verini şifreleme işlemini ECB modunda 5.95 mikro saniyede, CFB modunda 6.26 mikro saniyede, CBC modunda 6.21 mikro saniyede, OFB modunda 6.25 mikro saniyede ve CTR modunda 6.41mikro saniyede gerçekleştirmiştir. Aes algoritması 128 bit anahtar uzunluğunda en performanslı şifreleme işlemini ECB modu uygulandığında veriyor.



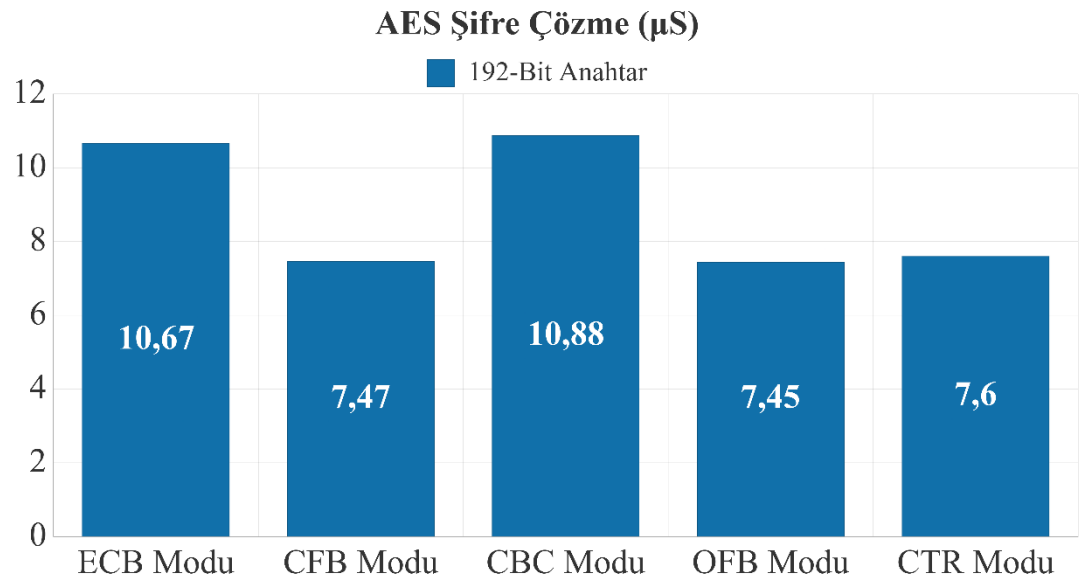
Şekil 3.10. Aes 128 Bit Anahtar Şifre Çözme

Aes algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifre çözme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir. Aes algoritması 128 bit anahtar kullanarak uygulanmıştır. Aes algoritması bir byte'lık verini şifre çözme işlemini ECB modunda 8.85 mikro saniyede, CFB modunda 6.29 mikro saniyede, CBC modunda 9.06 mikro saniyede, OFB modunda 6.26 mikro saniyede ve CTR modunda 6.41 mikro saniyede gerçekleştirmiştir. Aes algoritması 128 bit anahtar uzunluğunda en performanslı şifre çözme işlemini OFB modu uygulandığında veriyor.



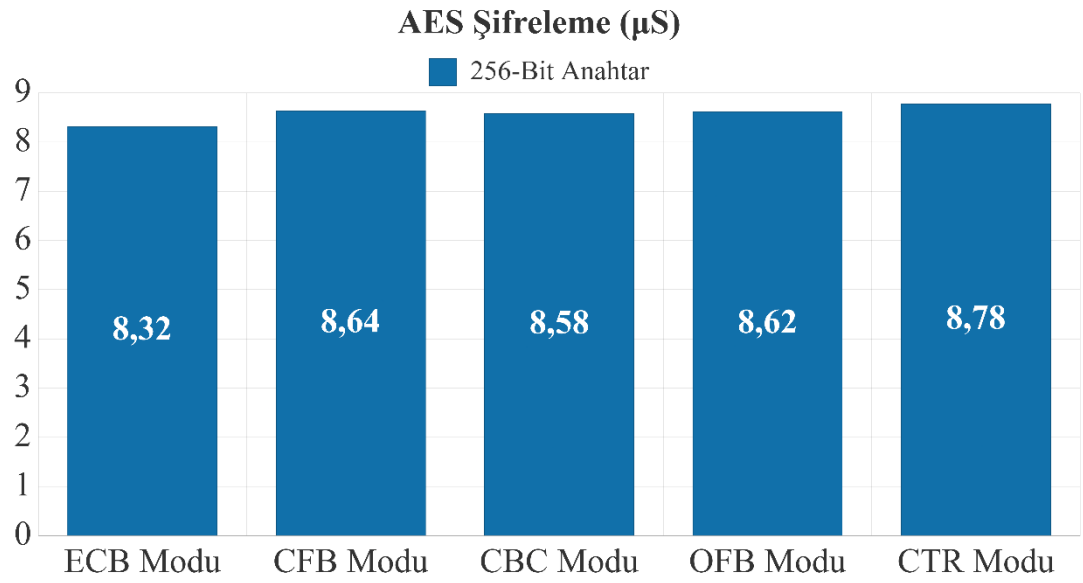
Şekil 3.11. Aes 192 Bit Anahtar Şifreleme

Aes algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifreleme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Aes algoritması 192 bit anahtar kullanarak uygulanmıştır. Aes algoritması bir byte'lık veriyi şifreleme işlemini ECB modunda 7.14 mikro saniyede, CFB modunda 7.45 mikro saniyede, CBC modunda 7.4 mikro saniyede, OFB modunda 7.44 mikro saniyede ve CTR modunda 7.59 mikro saniyede gerçekleştirmiştir. Aes algoritması 192 bit anahtar uzunluğunda en performanslı şifreleme işlemini ECB modu uygulandığında veriyor.



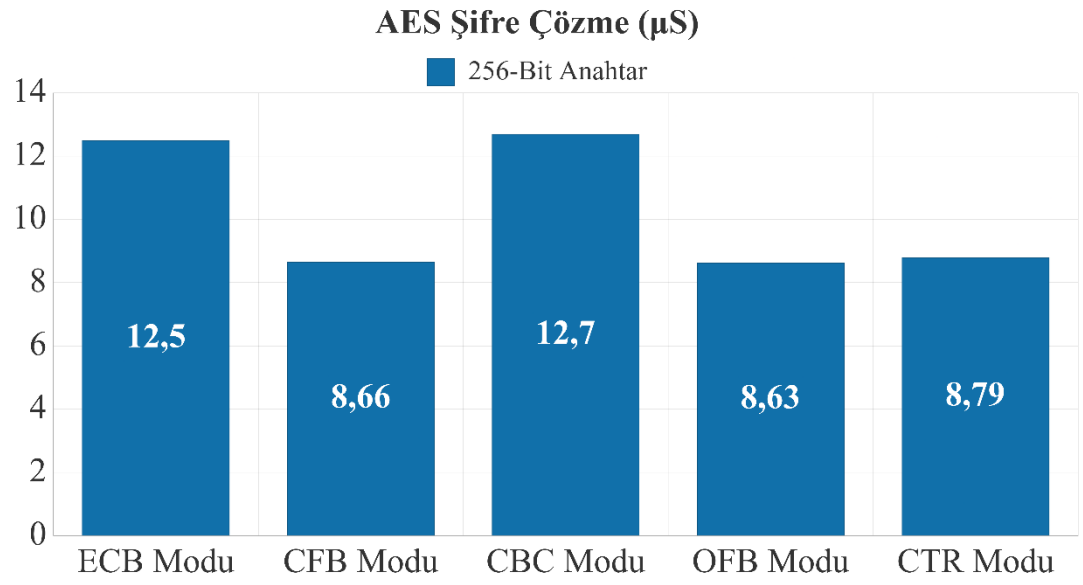
Şekil 3.12. Aes 192 Bit Anahtar Şifre Çözme

Aes algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifre çözme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Aes algoritması 192 bit anahtar kullanarak uygulanmıştır. Aes algoritması bir byte'lık verini şifre çözme işlemini ECB modunda 10.67 mikro saniyede, CFB modunda 7.47 mikro saniyede, CBC modunda 10.88 mikro saniyede, OFB modunda 7.45 mikro saniyede ve CTR modunda 7.6 mikro saniyede gerçekleştirmiştir. Aes algoritması 128 bit anahtar uzunluğunda en performanslı şifre çözme işlemini CFB modu uygulandığında veriyor.



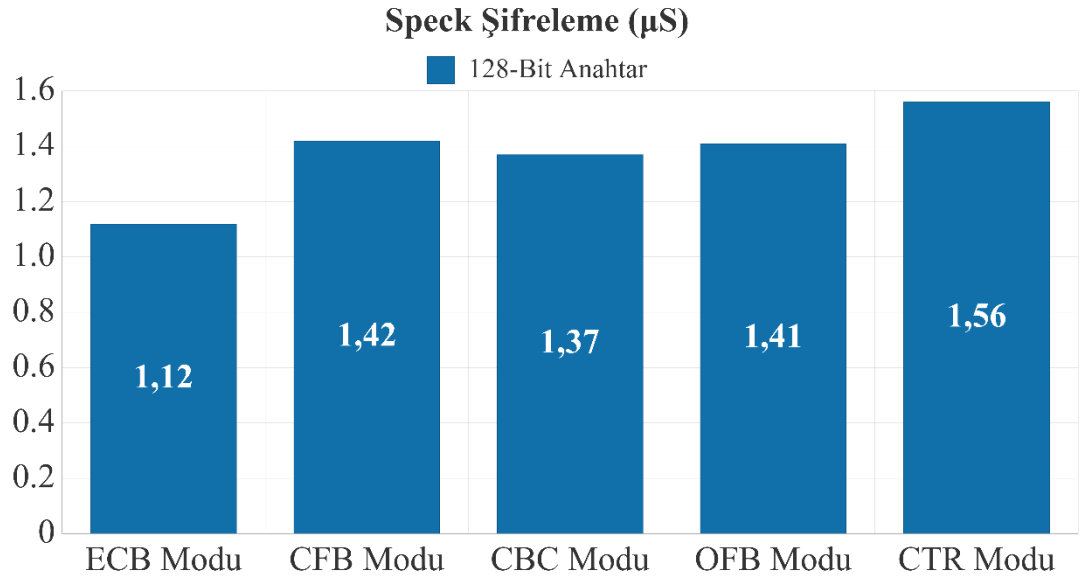
Şekil 3.13. Aes 256 Bit Anahtar Şifreleme

Aes algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifreleme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Aes algoritması 256 bit anahtar kullanarak uygulanmıştır. Aes algoritması bir byte'lık verini şifreleme işlemini ECB modunda 8.32 mikro saniyede, CFB modunda 8.64 mikro saniyede, CBC modunda 8.58 mikro saniyede, OFB modunda 8.62 mikro saniyede ve CTR modunda 8.78 mikro saniyede gerçekleştirmiştir. Aes algoritması 256 bit anahtar uzunluğunda en performanslı şifreleme işlemini ECB modu uygulandığında veriyor.



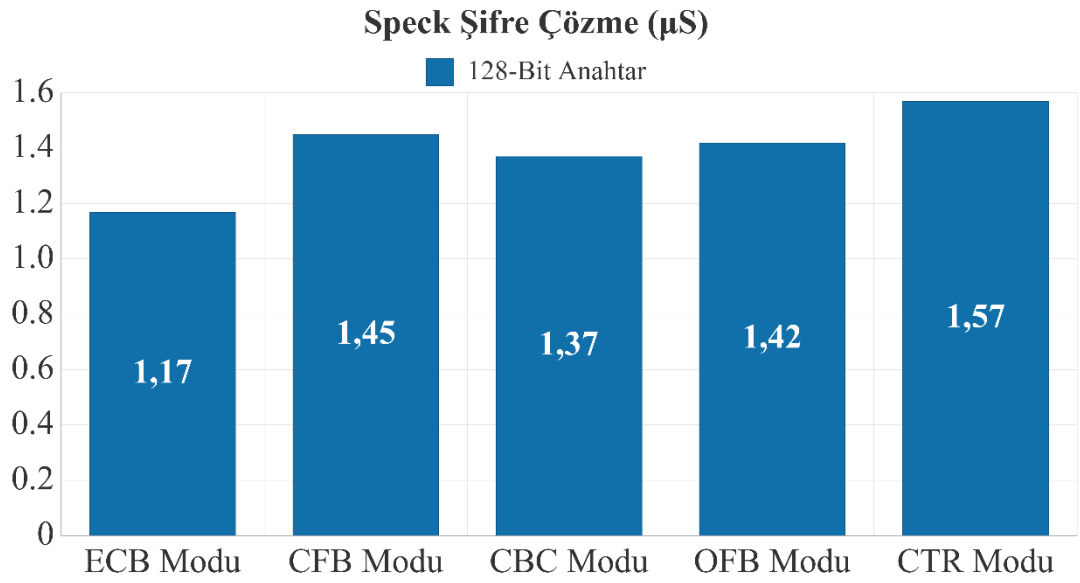
Şekil 3.14. Aes 256 Bit Anahtar Şifre Çözme

Aes algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifre çözme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Aes algoritması 256 bit anahtar kullanarak uygulanmıştır. Aes algoritması bir byte'lık verini şifre çözme işlemini ECB modunda 12.5 mikro saniyede, CFB modunda 8.66 mikro saniyede, CBC modunda 12.7 mikro saniyede, OFB modunda 8.63 mikro saniyede ve CTR modunda 8.79 mikro saniyede gerçekleştirmiştir. Aes algoritması 128 bit anahtar uzunluğunda en performanslı şifre çözme işlemini OFB modu uygulandığında veriyor.



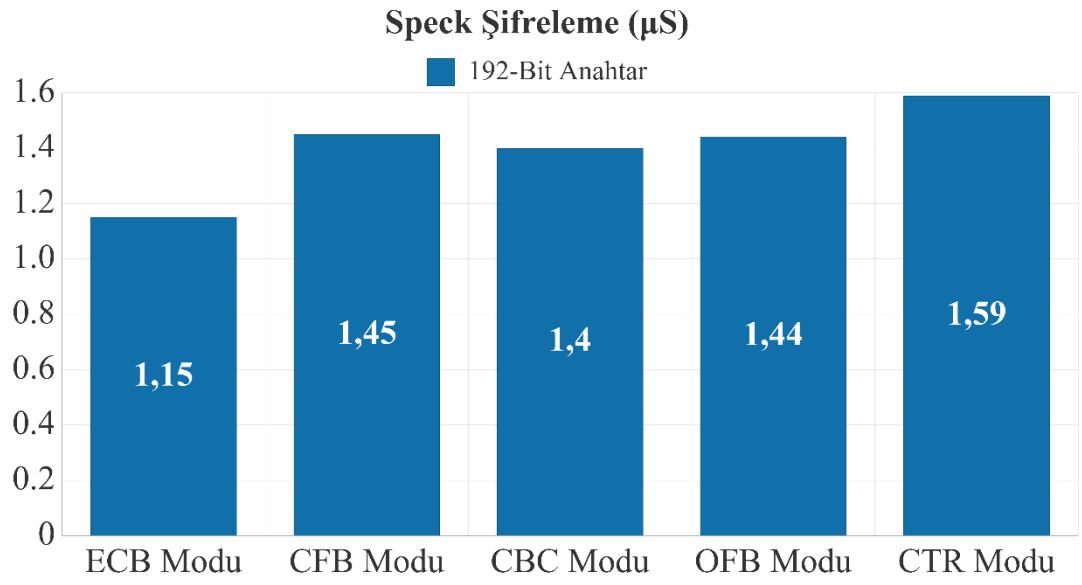
Şekil 3.15. Speck 128 Bit Anahtar Şifreleme

Speck algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifreleme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Speck algoritması 128 bit anahtar kullanarak uygulanmıştır. Speck algoritması bir byte'lık verini şifreleme işlemini ECB modunda 1.12 mikro saniyede, CFB modunda 1.42 mikro saniyede, CBC modunda 1.37 mikro saniyede, OFB modunda 1.41 mikro saniyede ve CTR modunda 1.57 mikro saniyede gerçekleştirmiştir. Speck algoritması 128 bit anahtar uzunluğunda en performanslı şifreleme işlemini ECB modu uygulandığında veriyor.



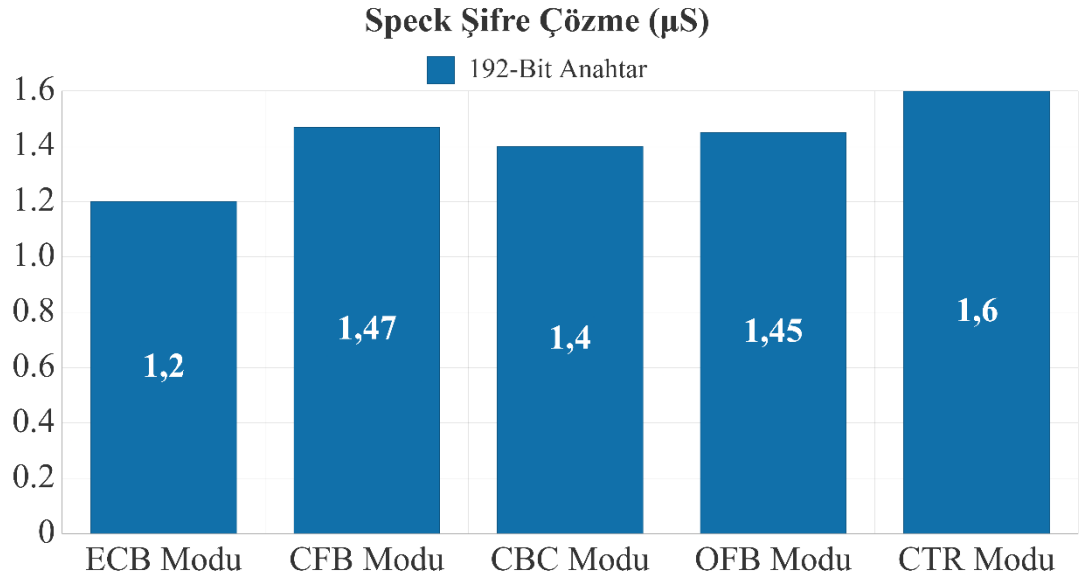
Şekil 3.16. Speck 128 Bit Anahtar Şifre Çözme

Speck algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifre çözme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Speck algoritması 128 bit anahtar kullanarak uygulanmıştır. Speck algoritması bir byte'lık verini şifre çözme işlemini ECB modunda 1.17 mikro saniyede, CFB modunda 1.45 mikro saniyede, CBC modunda 1.37 mikro saniyede, OFB modunda 1.42 mikro saniyede ve CTR modunda 1.57 mikro saniyede gerçekleştirmiştir. Speck algoritması 128 bit anahtar uzunluğunda en performanslı şifre çözme işlemini ECB modu uygulandığında veriyor.



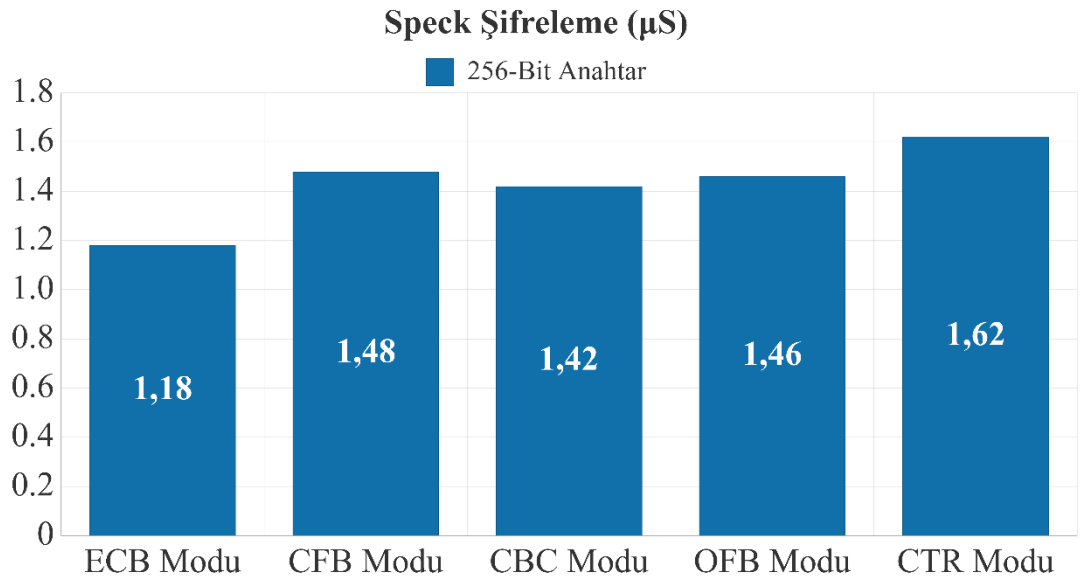
Şekil 3.17. Speck 192 Bit Anahtar Şifreleme

Speck algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifreleme işlemi gerçekleştirdiğimizde şekildeki grafik üretilmiştir. Speck algoritması 192 bit anahtar kullanarak uygulanmıştır. Speck algoritması bir byte'lık verini şifreleme işlemini ECB modunda 1.15 mikro saniyede, CFB modunda 1.45 mikro saniyede, CBC modunda 1.4 mikro saniyede, OFB modunda 1.44 mikro saniyede ve CTR modunda 1.59 mikro saniyede gerçekleştirmiştir. Speck algoritması 192 bit anahtar uzunluğunda en performanslı şifreleme işlemini ECB modu uygulandığında veriyor.



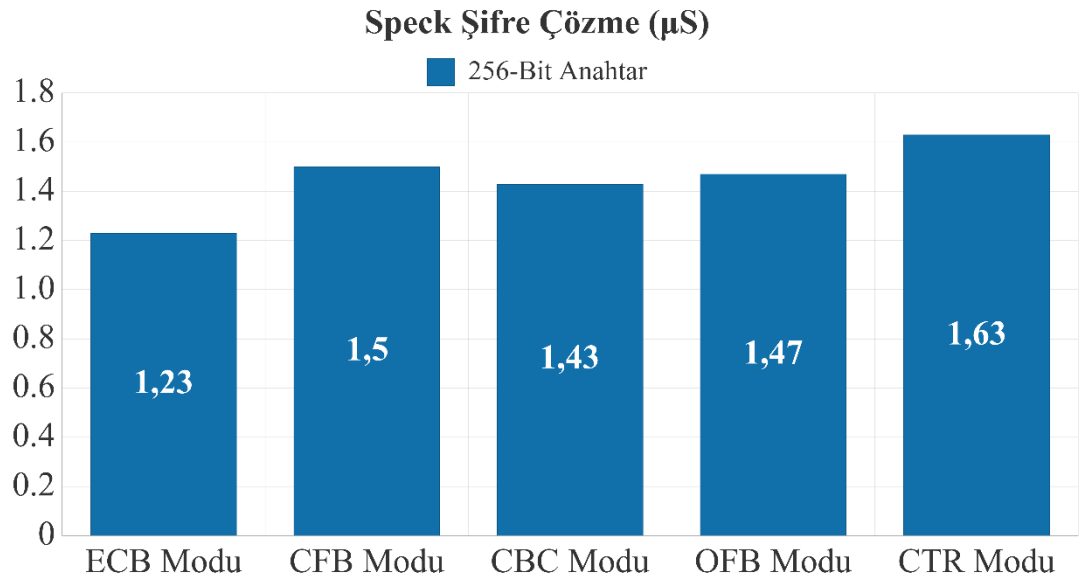
Şekil 3.18. Speck 192 Bit Anahtar Şifre Çözme

Speck algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifre çözme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir. Speck algoritması 192 bit anahtar kullanarak uygulanmıştır. Speck algoritması bir byte'lık verini şifre çözme işlemini ECB modunda 1.2 mikro saniyede, CFB modunda 1.47 mikro saniyede, CBC modunda 1.4 mikro saniyede, OFB modunda 1.45 mikro saniyede ve CTR modunda 1.6 mikro saniyede gerçekleştirmiştir. Speck algoritması 192 bit anahtar uzunluğunda en performanslı şifre çözme işlemini ECB modu uygulandığında veriyor.



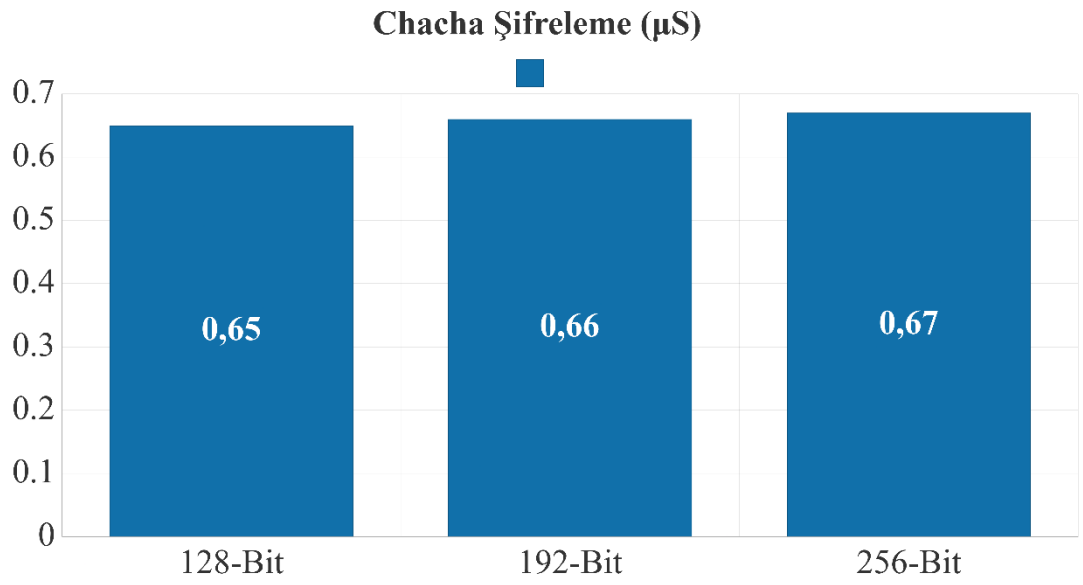
Şekil 3.19. Speck 256 Bit Anahtar Şifreleme

Speck algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifreleme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir. Speck algoritması 256 bit anahtar kullanarak uygulanmıştır. Speck algoritması bir byte'lık verini şifreleme işlemini ECB modunda 1.18 mikro saniyede, CFB modunda 1.48 mikro saniyede, CBC modunda 1.42 mikro saniyede, OFB modunda 1.46 mikro saniyede ve CTR modunda 1.62 mikro saniyede gerçekleştirmiştir. Speck algoritması 256 bit anahtar uzunluğunda en performanslı şifreleme işlemini ECB modu uygulandığında veriyor.



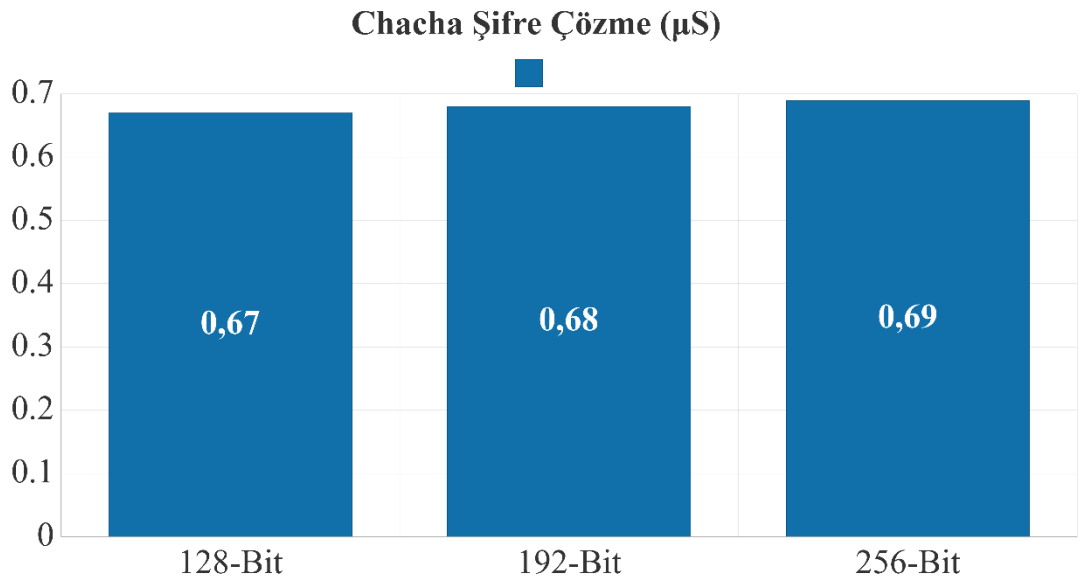
Şekil 3.20. Speck 256 Bit Anahtar Şifre Çözme

Speck algoritmasıyla ECB, CFB, CBC, OFB ve CTR modları ile şifre çözme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir. Speck algoritması 256 bit anahtar kullanarak uygulanmıştır. Speck algoritması bir byte'lık verini şifre çözme işlemini ECB modunda 1.23 mikro saniyede, CFB modunda 1.5 mikro saniyede, CBC modunda 1.43 mikro saniyede, OFB modunda 1.47 mikro saniyede ve CTR modunda 1.63 mikro saniyede gerçekleştirmiştir. Speck algoritması 256 bit anahtar uzunluğunda en performanslı şifre çözme işlemini ECB modu uygulandığında veriyor.



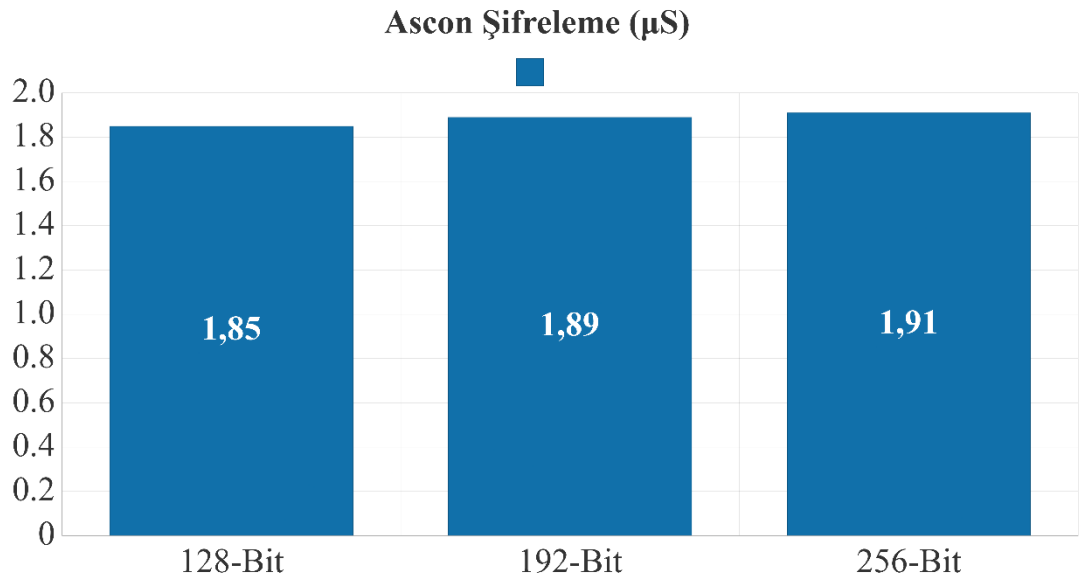
Şekil 3.21. Chacha 128, 192, 256 Bit Anahtar Şifreleme

Chacha algoritmasıyla 128 bit, 192 bit ve 256 bit anahtar ile şifreleme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir Chacha algoritması bir byte lık verini şifreleme işlemini 128 bitlik anahtar ile 0.65 mikro saniyede, 192 bitlik anahtar ile 0.66 mikro saniyede ve 256 bitlik anahtar ile 0.67 mikro saniyede gerçekleştirmiştir. Chacha algoritması 128 bit, 192 bit ve 256 bit anahtar uzunlukları arasında en performanslı şifreleme işlemini 128 bit anahtar uygulandığında veriyor.



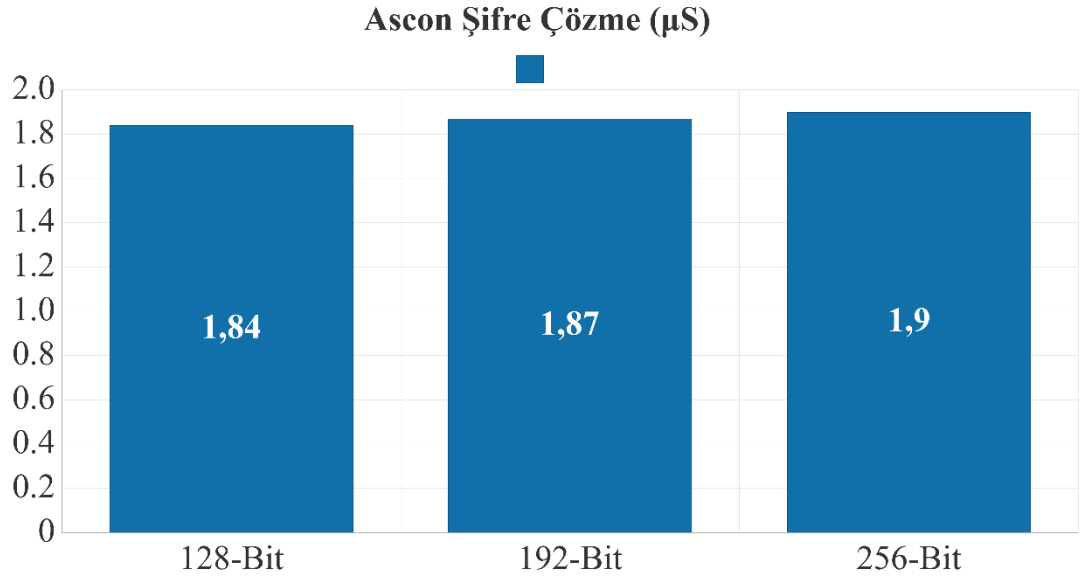
Şekil 3.22. Chacha 128, 192, 256 Bit Anahtar Şifre Çözme

Chacha algoritmasıyla 128 bit, 192 bit ve 256 bit anahtar ile şifre çözme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir Chacha algoritması bir byte lık verini şifre çözme işlemini 128 bitlik anahtar ile 0.67 mikro saniyede, 192 bitlik anahtar ile 0.68 mikro saniyede ve 256 bitlik anahtar ile 0.69 mikro saniyede gerçekleştirmiştir. Chacha algoritması 128 bit, 192 bit ve 256 bit anahtar uzunlukları arasında en performanslı şifre çözme işlemini 128 bit anahtar uygulandığında veriyor.



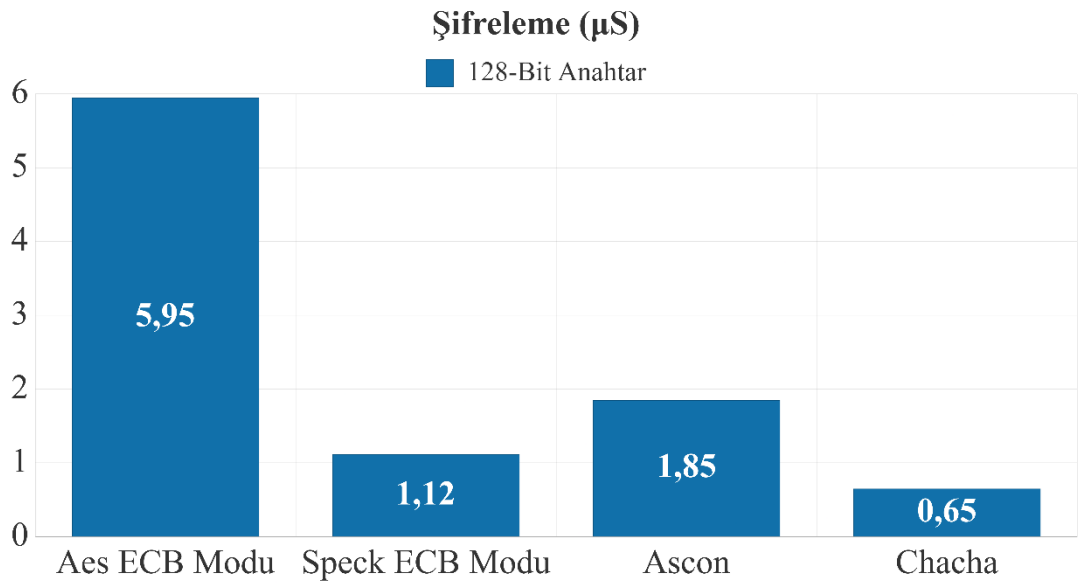
Şekil 3.23. Ascon 128, 192, 256 Bit Anahtar Şifreleme

Ascon algoritmasıyla 128 bit, 192 bit ve 256 bit anahtar ile şifreleme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir Ascon algoritması bir byte'lık verini şifreleme işlemini 128 bitlik anahtar ile 1.85 mikro saniyede, 192 bitlik anahtar ile 1.89 mikro saniyede ve 256 bitlik anahtar ile 1.91 mikro saniyede gerçekleştirmiştir. Ascon algoritması 128 bit, 192 bit ve 256 bit anahtar uzunlukları arasında en performanslı şifreleme işlemini 128 bit anahtar uygulandığında veriyor.



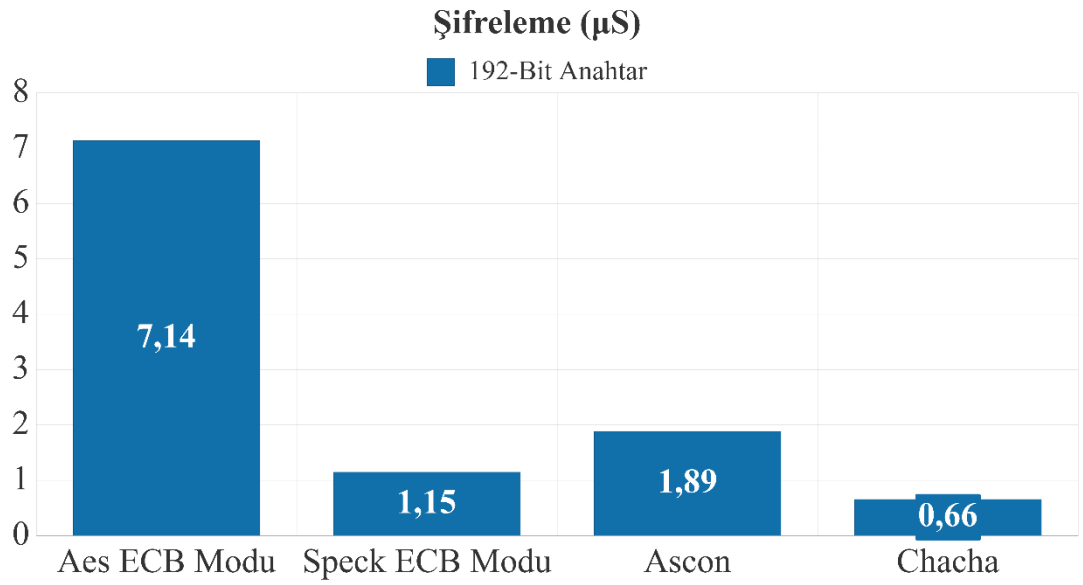
Şekil 3.24. Ascon 128, 192, 256 Bit Anahtar Şifre Çözme

Chacha algoritmasıyla 128 bit, 192 bit ve 256 bit anahtar ile şifre çözme işlemi gerçekleştirdiğimizde Şekildeki grafik üretilmiştir Chacha algoritması bir byte`lık verini şifre çözme işlemini 128 bitlik anahtar ile 1.84 mikro saniyede, 192 bitlik anahtar ile 1.87 mikro saniyede ve 256 bitlik anahtar ile 1.9 mikro saniyede gerçekleştirmiştir. Chacha algoritması 128 bit, 192 bit ve 256 bit anahtar uzunlukları arasında en performanslı şifre çözme işlemini 128 bit anahtar uygulandığında veriyor.



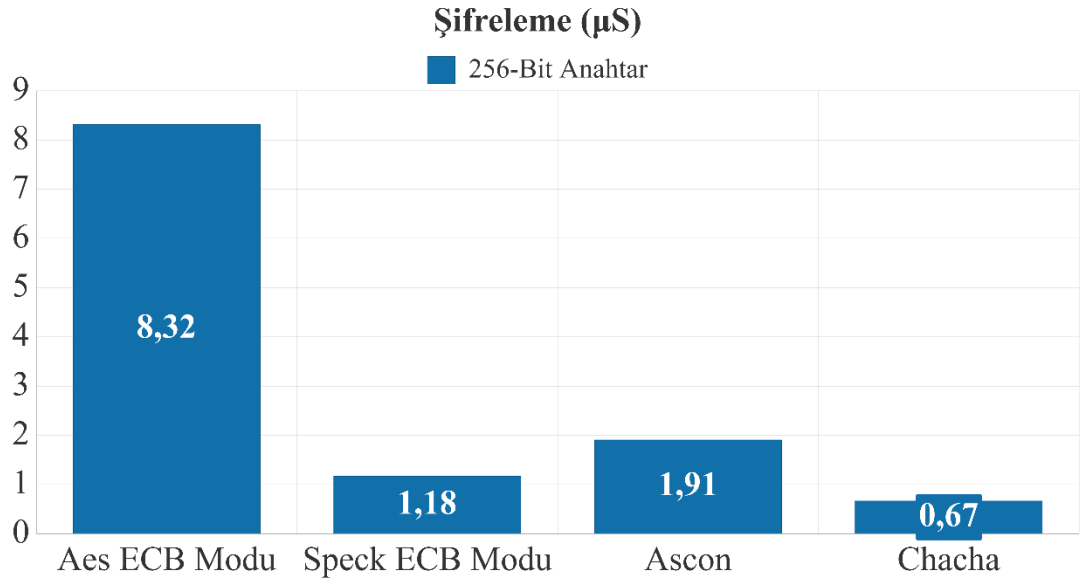
Şekil 3.25. 128 Bit Anahtar Şifreleme

128 bitlik anahtar kullanılarak Aes ve Speck algoritmaları ile gerçekleştirilen şifreleme işlemlerinde Aes algoritması için ECB, Speck algoritması için ECB modu en performanslı moddur. Dört şifreleme algoritması karşılaştırıldığında Chacha algoritması bir byte'lık veriyi 0.65 mikro saniyede şifrelemiştir. Chacha algoritması 128 bit anahtar ile şifreleme işleminde en performanslı şifrelemeyi gerçekleştirmiştir.



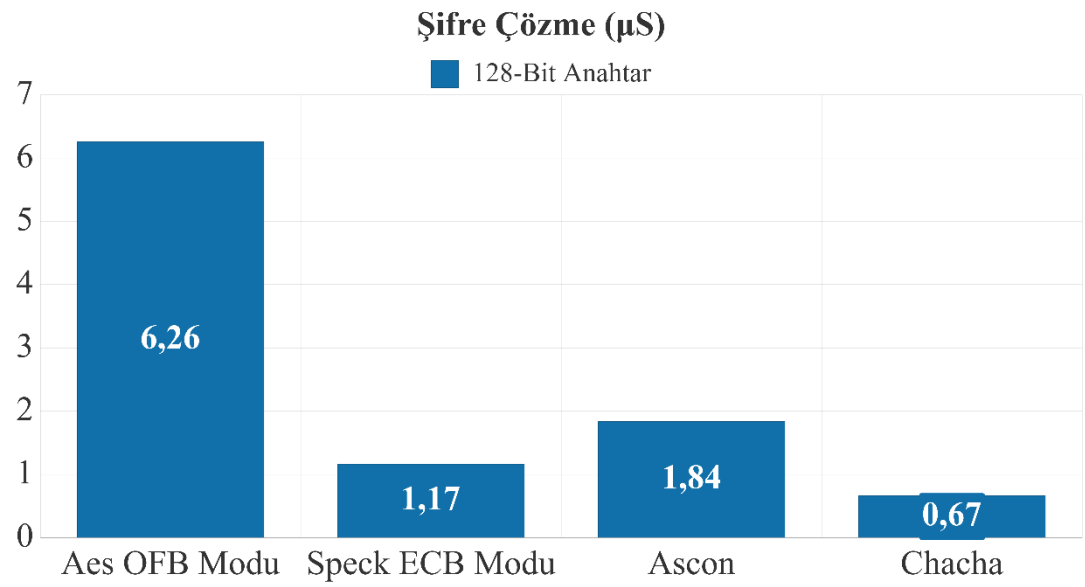
Şekil 3.26. 192 Bit Anahtar Şifreleme

192 bitlik anahtar kullanılarak Aes ve Speck algoritmaları ile gerçekleştirilen şifreleme işlemlerinde Aes algoritması için ECB, Speck algoritması için ECB modu en performanslı moddur. Dört şifreleme algoritması karşılaştırıldığında Chacha algoritması bir byte'lık veriyi 0.66 mikro saniyede şifrelemiştir. Chacha algoritması 192 bit anahtar ile şifreleme işleminde en performanslı şifrelemeyi gerçekleştirmiştir.



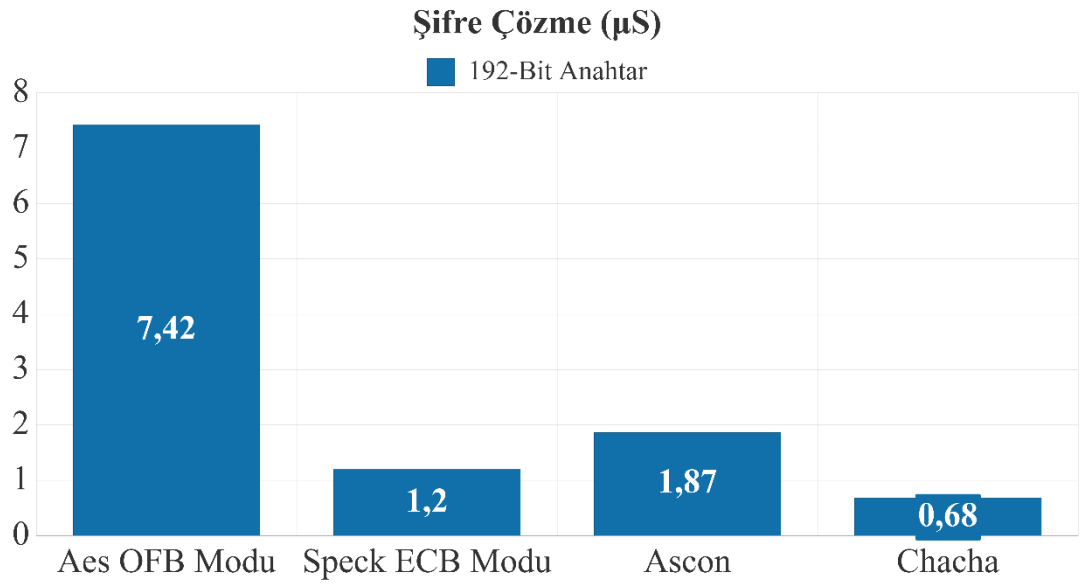
Şekil 3.27. 256 Bit Anahtar Şifreleme

256 bitlik anahtar kullanılarak Aes ve Speck algoritmaları ile gerçekleştirilen şifreleme işlemlerinde Aes algoritması için ECB, Speck algoritması için ECB modu en performanslı moddur. Dört şifreleme algoritması karşılaştırıldığında Chacha algoritması bir byte'lık veriyi 0.67 mikro saniyede şifrelemiştir. Chacha algoritması 256 bit anahtar ile şifreleme işleminde en performanslı şifrelemeyi gerçekleştirmiştir.



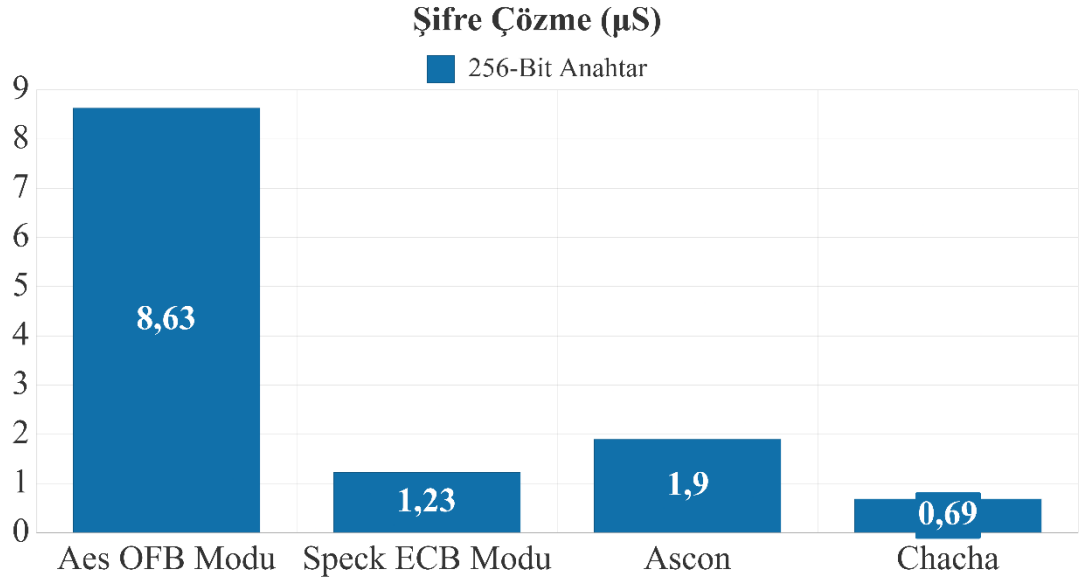
Şekil 3.28. 128 Bit Anahtar Şifre Çözme

128 bitlik anahtar kullanılarak Aes ve Speck algoritmaları ile gerçekleştirilen şifre çözme işlemlerinde Aes algoritması için OFB, Speck algoritması için ECB modu en performanslı moddur. Dört şifreleme algoritması karşılaştırıldığında Chacha algoritması bir byte'lık veriyi 0.67 mikro saniyede şifrelemiştir. Chacha algoritması 128 bit anahtar ile şifreleme işleminde en performanslı şifre çözmeyi gerçekleştirmiştir.



Şekil 3.29. 192 Bit Anahtar Şifre Çözme

192 bitlik anahtar kullanılarak Aes ve Speck algoritmaları ile gerçekleştirilen şifre çözme işlemlerinde Aes algoritması için OFB, Speck algoritması için ECB modu en performanslı moddur. Dört şifreleme algoritması karşılaştırıldığında Chacha algoritması bir byte'lık veriyi 0.68 mikro saniyede şifre çözmüştür. Chacha algoritması 192 bit anahtar ile şifreleme işleminde en performanslı şifre çözmeyi gerçekleştirmiştir.



Şekil 3.30. 256 Bit Anahtar Şifre Çözme

256 bitlik anahtar kullanılarak Aes ve Speck algoritmaları ile gerçekleştirilen şifreleme işlemlerinde Aes algoritması için OFB, Speck algoritması için ECB modu en performanslı moddur. Dört şifreleme algoritması karşılaştırıldığında Chacha algoritması bir byte'lık veriyi 0.69 mikro saniyede şifre çözmüştür. Chacha algoritması 256 bit anahtar ile şifreleme işleminde en performanslı şifre çözmeyi gerçekleştirmiştir.

4. SONUÇ

Akıllı ev cihazları düşük güç ve düşük performans gerektiren görevler için tasarlanmışlardır. Bu yüzden veri şifreleme işleminde geleneksel şifreleme algoritmaları yerine hafif sıklet şifreleme algoritmaları geliştirilmiştir.

Bu çalışmada hafif sıklet şifreleme algoritmalarının performans karşılaştırılması yapılmıştır. Aes algoritması kullanılarak geleneksel şifreleme algoritmalarının NodeMCU cihazı üzerinde hafif sıklet şifreleme algoritmalarına göre performans farkları incelenmiştir.

Tablo 4.1. Genel Şifreleme ve Şifre Çözme Performans Sonuçları

Konfigürasyon	**Algoritma**							
Anahtar Uzunluğu	**Şifreleme (µS)**				**Şifre Çözme (µS)**			
	Ascon	**Chacha**	**Aes (Mod)**	**Speck (Mod)**	**Ascon**	**Chacha**	**Aes (Mod)**	**Speck (Mod)**
128 Bit	1,85	**0,65**	5,95 (ECB)	1,12 (ECB)	1,84	**0,67**	6,26 (OFB)	1,17 (ECB)
192 Bit	1,89	**0,66**	7,14 (ECB)	1,15 (ECB)	1,87	**0,68**	7,42 (OFB)	1,2 (ECB)
256 Bit	1,91	**0,67**	8,32 (ECB)	1,18 (ECB)	1,9	**0,69**	8,63 (OFB)	1,23 (ECB)

Çalışmanın sonucu olarak Tablo 4.1.'de belirtildiği gibi geleneksel şifreleme algoritmalarından olan Aes şifreleme algoritmasının performansının hafif sıklet şifreleme algoritmaların performanslarına oranla 4-5 kata daha düşük olduğu verisine ulaşılmıştır. 128 bit, 192 bit ve 256 bit anahtar uzunlukları arasında yapılan performans ölçümlerinde şifreleme algoritmalarının en yüksek performansı 128 bit anahtar kullanılarak yapılan şifreleme işlemlerinde görülmüştür. En düşük performans ise 256 bit anahtar uzunluğunun uygulandığı şifreleme işlemlerinde görülmüştür. Bu verilerden yola çıkarak anahtar boyutunun şifreleme performansına etkisinin olduğu sonucuna varılmıştır. Şifreleme performansının şifreleme anahtarının uzunluğu ile ters orantılı olduğu sonucuna varılmıştır.

Aes şifreleme algoritması 128 bit anahtar kullanıldığında en performanslı şifreleme işlemini ECB moduyla gerçekleştirmiştir. Aes şifreleme algoritması 128 bit anahtar

kullanıldığında en performanslı şifre çözme işlemini OFB moduyla gerçekleştirmiştir. Aes şifreleme algoritması 192 bit anahtar kullanıldığında en performanslı şifreleme işlemini ECB moduyla gerçekleştirmiştir. Aes şifreleme algoritması 192 bit anahtar kullanıldığında en performanslı şifre çözme işlemini CFB moduyla gerçekleştirmiştir.

Aes şifreleme algoritması 256 bit anahtar kullanıldığında en performanslı şifreleme işlemini ECB moduyla gerçekleştirmiştir. Aes şifreleme algoritması 256 bit anahtar kullanıldığında en performanslı şifre çözme işlemini OFB moduyla gerçekleştirmiştir.

Speck şifreleme algoritması 128 bit anahtar kullanıldığında en performanslı şifreleme işlemini ECB moduyla gerçekleştirmiştir. Speck şifreleme algoritması 128 bit anahtar kullanıldığında en performanslı şifre çözme işlemini ECB moduyla gerçekleştirmiştir. Speck şifreleme algoritması 192 bit anahtar kullanıldığında en performanslı şifreleme işlemini ECB moduyla gerçekleştirmiştir. Speck şifreleme algoritması 192 bit anahtar kullanıldığında en performanslı şifre çözme işlemini ECB moduyla gerçekleştirmiştir. Speck şifreleme algoritması 256 bit anahtar kullanıldığında en performanslı şifreleme işlemini ECB moduyla gerçekleştirmiştir. Speck şifreleme algoritması 256 bit anahtar kullanıldığında en performanslı şifre çözme işlemini ECB moduyla gerçekleştirmiştir.

Şifreleme algoritmaların genel performans değerleri incelendiğinde 128 bit, 192 bit ve 256 bit anahtar uzunluğunda, şifreleme ve şifre çözme hızında Chacha şifreleme algoritmasın en performanslı işlem yaptığı sonucuna varılmıştır.

Akıllı ev cihazları arasında veri iletiminde düşük veri gecikmesiyle veri şifreleme işlemlerinde Chacha şifreleme algoritmasının uygulanması Aes, Speck ve Chacha algoritmalarına oranla daha verimli sonuç verecektir.

KAYNAKLAR

Abdul Hussien, F. T., Rahma, A. M. S., ve Abdul Wahab, H. B. (2021). A Secure Environment Using a New Lightweight AES Encryption Algorithm for E-Commerce Websites. *Security and Communication Networks*, (6), 1-15. https://doi.org/10.1155/2021/9961172

NodeMCU Teknik Özellikleri- Arduino Destek. (2022, Şubat 12). https://arduinodestek.com/tag/nodemcu-teknik-ozellikleri/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Akıllı ev sistemleri nedir? | TeoremEnerji | 2023., https://teoremenerji.com.tr/akilli-ev-sistemleri-nedir/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Akıllı Ev Teknolojileri- INTERPOINT Teknoloji. (2018, Ekim 17). https://e-akilliev.com/akilli-ev-teknolojileri/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Ana Sayfa | Lora Teknolojisi Nedir? | Nerelerde Kullanılır | esgsistem. (2020). Esgsistem | Lora IOT Sistemi., https://loranedir.com/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Aslan B., Sakallı M. T., (2008). S-kutularının Kriptografik Özellikleri , Eleco

International Conference on Electrical and Electronics Eng., Bursa, Türkiye.

Avcı, İ. (2022). Akıllı Evlerde IoT Teknolojileri ve Siber Güvenlik. *European Journal of Science and Technology*, (34), 226-233. https://doi.org/10.31590/ejosat.1080228

Ayşe, Ö., ve Sepanta, N. (2017). Akıllı Ev Sistemlerinde Kullanılan Yöntemlerin Farkları, Avantajları ve Dezavantajları, *İstanbul Aydın Üniversitesi Dergisi,* 9(4), 115-125.

Beaulieu, R., Treatman-Clark, S., Shors, D., Weeks, B., Smith, J., ve Wingers, L. (2015). The SIMON and SPECK lightweight block cIPhers. *Proceedings-Design Automation Conference,* (175), 1-6. https://doi.org/10.1145/2744769.2747946

Çolak, B. (2020, Haziran 2). Akıllı Evler ve Güvenlik. Deep Learning Türkiye. https://medium.com/deep-learning-turkiye/ak%C4%B1ll%C4%B1-evler-ve-g%C3%BCvenlik-9250735e5b9f adresinden 23 Aralık 2023 tarihinde alınmıştır.

Crypto 101 – Şifreleme Operasyonu Modları – ECB, CBC, OFB – Mehmet INCE | Information Security. (2015, Eylül 1). https://www.mehmetince.net/crypto-101-5-sifreleme-operasyonu-modlari-ecb-cbc-ofb/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Çavuşoğlu, Ü., ve Al-sanabani, H. (2019). Hafif Sıklet Şifreleme Algoritmalarının Performans Karşılaştırması. *Sakarya University Journal of Computer and Information Sciences*, *2*(3), 158–169. https://doi.org/10.35377/saucis.02.03.648493

DB Browser for SQLite. (2019). rowser.org. https://sqlitebrowser.org/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Dobraunig, C., Eichlseder, M., Mendel, F., ve Schläffer, M. (2021). Ascon v1.2: Lightweight Authenticated Encryption and Hashing. *Journal of Cryptology*, *34*(3). https://doi.org/10.1007/s00145-021-09398-9

Editor, C. C. (2020). Authentication Tag- Glossary | CSRC. Csrc.nist.gov. , https://csrc.nist.gov/glossary/term/authentication_tag adresinden 23 Aralık 2023 tarihinde alınmıştır.

ÖZDENİZ, D. (2020). Evler Ne Kadar Akıllı Olmalı, Tasarım ve Uygulamadaki Problemler, Çözüm Önerileri.

Eray, S. S. (2020). Anadolu Geleneksel Evlerinin İç Mekân Kurgusu. *International Journal of Mardin Studies (IJMS)*, *2*(1), 27-39.

Güğül, G. N., ve Sarıtaş, M. (2020). Akıllı Ev Sistemleri ve Uygulaması. *DPÜ Fen Bilimleri Enstitüsü Dergisi,* 025, 49-60.

Evlerde kullanılan akıllı cihazlar ciddi güvenlik riskleri taşıyor olabilir. (2023, Nisan 29). Haberler. https://www.haberler.com/ekonomi/evlerde-kullanilan-akilli-cihazlar-ciddi-guvenlik-15837970-haberi/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Göğebakan, Y. (2015). Karakteristik Bir Değer Olan Geleneksel Türk Evi'nin Oluşumunu Belirleyen Unsurlar ve Bu Evlerin Genel Özellikleri, *İnönü University Journal of Culture and Art*, *1*(1), 41-55.

HTTP Nedir? Ne İşe Yarar? HTTPS ile Farkı Nedir? | Pomelo Soft - Pomelo Soft. (2020). Www.pomelosoft.com. https://www.pomelosoft.com/blog/http-nedir adresinden 23 Aralık 2023 tarihinde alınmıştır.

İlgazi. (2022, Aralık 19). Rfid Nedir ve Nasıl Çalışır? - 2023. İlgazi Rfid Etiket Izmir Istanbul. https://www.ilgazi.com/blog/rfid/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Inohom Akıllı Ev Sistemleri. (2023). Inohom.com. https://www.inohom.com/akilli-ev-nedir adresinden 23 Aralık 2023 tarihinde alınmıştır.

IoT güvenliği ev ağınız için neden önemlidir? (2023, Nisan 19). www.kaspersky.com.tr. https://www.kaspersky.com.tr/resource-center/threats/secure-iot-devices-on-your-home-network adresinden 23 Aralık 2023 tarihinde alınmıştır.

Iot'de Veri Tazeliğinin Önemi | Ankaref. (2020). Ankaref.com. , https://ankaref.com/blog/iotde-veri-tazeliginin-onemi-39 adresinden 23 Aralık 2023 tarihinde alınmıştır.

İlkbahar, F., Ünal, Ş., Karakaya, A. T., ve Eren, B. (2021). Akıllı Ev Sistemleri Üzerine Bir Model Önerisi. *AJIT-e Online Academic Journal of Information Technology*, *12*(45), 90–105. https://doi.org/10.5824/ajite.2021.02.005.x

Kandır, M. O., Yolaçan, E., ve Işık, Ş. (2022). Security Of the Internet of Things: Home Network Security Review and Evaluation. *Uludağ University Journal of The Faculty of Engineering*, *27*(2), 803–816. https://doi.org/10.17482/uumfd.1068960

Katagi, M., & Moriai, S. (2020). Lightweight Cryptography for the Internet of Things. http://www.ecrypt.eu.org/stream/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

KAZANCI, M. İ. (2021, Ağustos 9). MQTT Nedir? Nasıl Kullanılır? ve SDKs. Medium. https://ikbalkazanc.medium.com/mqtt-nedir-nas%C4%B1l-kullan%C4%B1l%C4%B1r-ve-sdks-1851d0f7a7fa adresinden 23 Aralık 2023 tarihinde alınmıştır.

Korkmaz, S. (2017, Aralık 20). Akıllı Ev Sistemi Nedir? Temel Özellikleri Nelerdir? Bilgiustam. Bilgiustam. https://www.bilgiustam.com/akilli-ev-sistemi-nedir-temel-ozellikleri-nelerdir/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Koçyiğit, Y., ve Sine, Ö. (2020). İnternet Üzerinden Kontrol Edilen Tam Otomasyonlu Akıllı Ev Sistemleri İçin Örnek Bir Uygulama. *DÜMF Mühendislik Dergisi, 11*(2), 521–532. https://doi.org/10.24012/dumf.635296

Mesleğim Hayatım. (2020). Mtegm.meb.gov.tr. https://mtegm.meb.gov.tr/kalfalik_ustalik_sinavlari/Dersler/guvenlik_sistemleri/akilli_ev_sistemleri.html adresinden 23 Aralık 2023 tarihinde alınmıştır.

Mills, M. (2022, Eylül 20). SQLite veritabanı yöneticisi için DB Tarayıcısını indirin ve kullanın | ITIGIC. Itigic.com. https://itigic.com/tr/download-and-use-db-browser-for-sqlite-database-manager/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

MQTT Nedir ve Nasıl Çalışır? (2020). IoT Türkiye | Türkiye'nin En Büyük Nesnelerin İnterneti Ekosistemi. , https://ioturkiye.com/2022/10/mqtt-nedir-ve-nasil-calisir/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Muhalha, L. A., & Alshawi, I. S. (2023). A Hybrid Modified Lightweight Algorithm For Achieving Data İntegrity And Confidentiality. *International Journal of Electrical and Computer Engineering*, 13(1), 833–841. https://doi.org/10.11591/ijece.v13i1.pp833-841

Musa, M. A., Schaefer, E. F., & Wedig, S. (2003). A Simplified Aes Algorithm And İts Linear And Differential Cryptanalyses. *Cryptologia*, 27(2), 148–177. https://doi.org/10.1080/0161-110391891838

Nesnelerin İnterneti (IoT) Nedir? (2020). Oracle.com. https://www.oracle.com/tr/internet-of-things/what-is-iot/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Nesnelerin İnterneti (IoT) Nedir? Neden Önemlidir? - Midas. (2022, Eylül 30). https://www.getmidas.com/blog/nesnelerin-interneti-nedir/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Nesnelerin İnterneti nedir? Tanım ve açıklama. (2023, Nisan 19). Www.kaspersky.com.tr. https://www.kaspersky.com.tr/resource-center/definitions/what-is-iot adresinden 23 Aralık 2023 tarihinde alınmıştır.

NFC Nedir, Nasıl Kullanılır? (2020). Türkiye Finans. , https://www.turkiyefinans.com.tr/tr-tr/blog/sayfalar/nfc-teknolojisi-nedir.aspx adresinden 23 Aralık 2023 tarihinde alınmıştır.

NFC ve IoT: Nesnelerin İnternetine Entegrasyon | Avukat Kartviziti. (2020). Www.avukatkartviziti.com. , https://www.avukatkartviziti.com/blog-NFC-ve-IoT-Nesnelerin-Internetine-Entegrasyon.html adresinden 23 Aralık 2023 tarihinde alınmıştır.

Nordquist, T. (2020). MQTT Explorer. MQTT Explorer. https://mqtt-explorer.com/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Kriptografi Yöntemleri- Asimetrik ve Simetrik Şifreleme Nedir? (2021, Eylül 12). Ofcskn. https://ofcskn.com/tr/kriptografi-yontemleri-nelerdir-asimetrik-ve-simetrik-sifreleme-nedir adresinden 23 Aralık 2023 tarihinde alınmıştır.

Panahi, U. (2020). *Nesnelerin interneti için hafif sıklet kriptoloji algoritmalarına dayalı güvenli haberleşme modeli tasarımı* [Doktora tezi] Sakarya Üniversitesi.

Pandey, R. (2023). Lightweight Symmetric Encryption and Attribute Based Encryption Method to Increase Information Safety in Wireless Sensor Network. *Journal of Cybersecurity and Information Management*, *10*(2), 47–56. https://doi.org/10.54216/JCIM.100205

RFID Nedir? RFID Etiketlerinin Avantajları Nelerdir? | ETMD. (2022, Şubat 18). https://www.etmd.org.tr/rfid-nedir-rfid-etiketlerinin-avantajlari-nelerdir/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

S-Kutuları (S-Boxes) – Bilgisayar Kavramları. (2008, Haziran 7). https://bilgisayarkavramlari.com/2008/06/07/s-kutulari-s-boxes/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Sadhukhan, R., Patranabis, S., Ghoshal, A., Mukhopadhyay, D., Saraswat, V., ve Ghosh, S. (2017). An Evaluation of Lightweight Block Ciphers for Resource-Constrained Applications: Area, Performance, and Security. *Journal of Hardware and Systems Security*, *1*(3), 203–218. https://doi.org/10.1007/s41635-017-0021-2

Şifreleme nedir ve siber güvenlik için neden önemlidir? (2023, Nisan 11). https://antivirus.com.tr/sifreleme-nedir-ve-siber-guvenlik-acisindan-neden-onemlidir/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

What is Stream Cipher? - Definition, Attacks, and More. (2020, Ocak 2). Computer Tech Reviews. https://www.computertechreviews.com/definition/stream-cipher/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

What is the purpose of an Authentication Tag in AEAD encryption schemes? (2020). Cryptography Stack Exchange., https://crypto.stackexchange.com/questions/67107/what-is-the-purpose-of-an-authentication-tag-in-aead-encryption-schemes adresinden 23 Aralık 2023 tarihinde alınmıştır.

WiFİ Nedir? Nasıl Çalışır? Özellikleri Nelerdir? - Elektrikde. (2020). Www.elektrikde.com. , https://www.elektrikde.com/wifi-nedir-nasil-calisir-ozellikleri-nelerdir-2/ adresinden 23 Aralık 2023 tarihinde alınmıştır.

Galois/Sayaç Modu. (2020, Nisan 19). Wikipedia.org; Wikimedia Foundation, Inc. https://tr.wikipedia.org/wiki/Galois/Saya%C3%A7_Modu adresinden 23 Aralık 2023 tarihinde alınmıştır.

Yumurtacı, M. ve Keçebaş, A. (2009, Mayıs, 13-15). Akıllı Ev Teknolojiler ve Otomasyon Sistemleri [Sözlü sunum]. 5. Uluslararası İleri Teknolojiler Sempozyumu, Afyon, Türkiye.

Z-Wave ve ZigBee Nedir? Nasıl Seçim Yapılmalıdır? (2020) https://www.elektrikport.com/makale-detay/z-wave-ve-zigbee-nedir-nasil-secim-yapilmalidir/21948#ad-image-0 adresinden 23 Aralık 2023 tarihinde alınmıştır.

ÖZGEÇMİŞ

Ad-Soyad : Ömer YEL

ÖĞRENİM DURUMU:

- **Lisans :** 2019, Sakarya Üniversitesi, Bilgisayar ve Bilişim Bilimleri Fakültesi, Bilgisayar Mühendisliği Bölümü

MESLEKİ DENEYİM:

- 2022–2023 yılları arasında UBFSoft yazılım şirketinde yazılım geliştirici olarak çalıştı.
- 2023 yılından beri Fintech Yazılım şirketinde yazılım geliştirici olarak çalışıyor.

TEZDEN TÜRETİLEN ESERLER:

- Yel, Ö., Eski, H. 2023. Akıllı Ev Sistemlerinin Haberleşmesinde Hafif Sıklet Şifreleme Algoritmalarının Performans Karşılaştırılması, *International Science and Technology Conference*, Roma, İtaly, ISSN-2146-7366.