T.C.
İSTANBUL SABAHATTİN ZAİM ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR BİLİMLERİ VE MÜHENDİSLİĞİ (%30 İngilizce)
BİLİM DALI

BLOK ZİNCİR TABANLI AKILLI SÖZLEŞMELER İLE NESNELERİN İNTERNETİ SİSTEMLERİNDE GÜVENLİ GÖRÜNTÜ PAYLAŞIMI

DOKTORA TEZİ

Burak AĞGÜL

İstanbul
Haziran-2025

T.C.
İSTANBUL SABAHATTİN ZAİM ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR BİLİMLERİ VE MÜHENDİSLİĞİ (%30 İngilizce)
BİLİM DALI

BLOK ZİNCİR TABANLI AKILLI SÖZLEŞMELER İLE NESNELERİN İNTERNETİ SİSTEMLERİNDE GÜVENLİ GÖRÜNTÜ PAYLAŞIMI

DOKTORA TEZİ

Burak AĞGÜL

Tez Danışmanı
Doç. Dr. Tayfun ACARER
Eş Danışmanı
Doç. Dr. Gökhan ERDEMİR

İstanbul
Haziran-2025

TEZ ONAY

Lisansüstü Eğitim Enstitüsü Müdürlüğüne,

Bu çalışma, jürimiz tarafından Bilgisayar Mühendisliği Anabilim Dalı, Bilgisayar Bilimleri ve Mühendisliği (30% İngilizce) Bilim Dalında DOKTORA TEZİ olarak kabul edilmiştir.

Danışman Doç. Dr. Tayfun ACARER

Üye Prof. Dr. Burhanettin CAN

Üye Dr. Öğr. Üyesi Hakan GENÇOĞLU

Üye Dr. Öğr. Üyesi Oktay DOĞANGÜN

Üye Dr. Öğr. Üyesi Erdal ALIMOVSKI

Onay

Yukarıdaki imzaların, adı geçen öğretim üyelerine ait olduğunu onaylarım.

Prof. Dr. Erhan İÇENER
Enstitü Müdürü

BİLİMSEL ETİK BİLDİRİMİ

Doktora tezi olarak hazırladığım **"BLOK ZİNCİR TABANLI AKILLI SÖZLEŞMELER İLE NESNELERİN İNTERNETİ SİSTEMLERİNDE GÜVENLİ GÖRÜNTÜ PAYLAŞIMI"** adlı çalışmanın öneri aşamasından sonuçlandığı aşamaya kadar geçen süreçte bilimsel etiğe ve akademik kurallara özenle uyduğumu, tez içindeki tüm bilgileri bilimsel ahlak ve gelenek çerçevesinde elde ettiğimi, tez yazım kurallarına uygun olarak hazırladığımı, bu çalışmamda doğrudan veya dolaylı olarak yaptığım her alıntıya kaynak gösterdiğimi ve yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu beyan ederim.

Burak AĞGÜL

ÖNSÖZ

Günümüzde, Nesnelerin İnterneti (IoT) sistemlerinin hızla yaygınlaşması ile birlikte, bu sistemlerden elde edilen veri çeşitliliği ve miktarı da artış göstermiştir. Özellikle gerçek zamanlı görüntü verilerinin güvenli ve verimli bir şekilde yönetilmesi, güncel teknolojik araştırmaların odak noktalarından biri haline gelmiştir.
Bu doğrultuda hazırlanan bu tez çalışmasında, IoT tabanlı görüntü verilerinin blok zincir teknolojisi kullanılarak güvenli ve bütünlüğü korunmuş şekilde iletilmesi amaçlanmıştır. Geliştirilen sistemde, Hyperledger Besu platformu ve Python ile geliştirilen akıllı sözleşmelerden yararlanılarak, veri bütünlüğünü sağlama, gecikmeyi azaltma ve kaynak tüketimini optimize etme hedeflenmiştir.

Çalışmanın her aşamasında, elde edilen bulguların akademik katkı sunması ve uygulama alanlarında değer yaratması hedeflenmiştir. Bu süreçte, karşılaşılan zorluklar ve elde edilen deneyimler, kişisel ve akademik gelişimime önemli katkılar sağlamıştır.

Çalışmamın başarıyla tamamlanabilmesi, başta danışmanlarım olmak üzere birçok kişinin desteği ve katkılarıyla mümkün olmuştur. Teşekkürlerimi ve takdirlerimi sunma fırsatı bulduğum Teşekkür bölümünde tüm katkı sağlayanlara ayrıca yer verilmiştir.

Bu vesileyle, bu çalışmanın ilgili literatüre ve uygulama alanlarına fayda sağlamasını temenni eder, tüm okuyuculara faydalı olmasını dilerim.

Burak AĞGÜL

Haziran-2025

TEŞEKKÜR

Bu araştırma süresince göstermiş olduğu değerli katkı ve rehberlik desteği için Danışmanım Doç. Dr. Tayfun ACARER' e en içten teşekkürlerimi sunarım. Kendisinin akademik yönlendirmeleri ve katkıları, bu tezin başarıyla tamamlanmasına önemli ölçüde katkı sağlamıştır.

Ayrıca, araştırma sürecimin her aşamasında yanımda olarak bilgi ve tecrübeleriyle çalışmanın gelişimine büyük katkılar sunan Eş Danışmanım Doç. Dr. Gökhan ERDEMİR' e de özel olarak teşekkür ederim. Kendisinin sürekli desteği, yol göstericiliği ve yapıcı önerileri, çalışmanın bilimsel derinliğini ve kalitesini artırmada belirleyici olmuştur.

Tez jüri üyeleri hocalarım Prof. Dr. Burhanettin CAN, Dr. Öğr. Üyesi Hakan GENÇOĞLU, Dr. Öğr. Üyesi Oktay DOĞANGÜN ve Dr. Öğr. Üyesi Erdal ALIMOVSKI hocalarıma zaman ayırarak sundukları değerli görüş, öneri ve katkılarından dolayı teşekkür ederim.

Araştırmanın gerçekleştirilmesine olanak sağlayan laboratuvar ve teknik altyapı desteği için İstanbul Sabahattin Zaim Üniversitesi Nükleer Algılayıcılar ve Robotik Uygulama ve Araştırma Merkezi (NAR) yönetimine ve her türlü desteğiyle yanımda olan değerli hocam Prof. Dr. Mustafa Nizamettin Erduran'a teşekkür ederim.

Bu süreçte desteklerini esirgemeyen tüm meslektaşlarıma ve arkadaşlarıma teşekkür ederim.

Son olarak, her zaman yanımda olan, sabır ve anlayışla bana destek veren aileme en derin teşekkürlerimi sunarım.

Burak AĞGÜL

Haziran-2025

ÖZET

BLOK ZİNCİR TABANLI AKILLI SÖZLEŞMELER İLE NESNELERİN İNTERNETİ SİSTEMLERİNDE GÜVENLİ GÖRÜNTÜ PAYLAŞIMI

Burak AĞGÜL

Doktora, Bilgisayar Bilimleri ve Mühendisliği (%30 İngilizce)

Tez Danışmanı: Doç. Dr. Tayfun ACARER

Eş Danışmanı: Doç. Dr. Gökhan ERDEMİR

Haziran, 2025 - 110 +XV Sayfa

Bu çalışmanın amacı, Nesnelerin İnterneti (IoT) ortamlarında gerçek zamanlı görüntü verisini blok zincir tabanlı bir sistem üzerinden güvenli ve bütünlüğü korunmuş olarak iletmektir. Araştırma, Hyperledger Besu platformu ve akıllı sözleşmeler kullanılarak IoT cihazlarından alınan görüntülerin güvenli aktarımına odaklanmaktadır. Bu kapsamda, TurtleBot ve Raspberry Pi donanımlarıyla dinamik ve statik görüntüler yakalanmış, veriler 128 KB'lik parçalara bölünerek verimli şekilde aktarılmış ve SHA-256 özet algoritması ile bütünlük doğrulanmıştır. Sistem, QBFT mutabakat algoritmasıyla yapılandırılmış dört düğümlü bir Hyperledger Besu ağında çalışmakta ve tüm kayıtların güvenilir biçimde saklanmasını sağlamaktadır. Deneysel sonuçlar, sistemin veri bütünlüğünü koruma, gerçek zamanlı ve gizlilik odaklı güvenli veri iletimi konularında başarılı olduğunu göstermiştir. Çalışma; araştırma problemi, literatür incelemesi, blok zincir ve IoT temelleri, yöntem ve uygulama, sistem tasarımı, performans değerlendirmesi, güvenlik analizi, tartışma ile sonuç ve gelecekteki çalışmalar olmak üzere dokuz bölümden oluşmaktadır.

Anahtar Kelimeler: Blok zincir, nesnelerin interneti, görüntü verisi aktarımı, Hyperledger Besu, akıllı sözleşmeler, veri bütünlüğü, mutabakat algoritmaları.

ABSTRACT

SECURE IMAGE SHARING INTERNET OF THINGS SYSTEMS WITH BLOCKCHAIN-BASED SMART CONTRACTS

Burak AĞGÜL

Ph. D. Computer Sciences and Engineering (30% English)

Supervisor: Assoc. Prof. Dr. Tayfun ACARER

Co-Supervisor: Assoc. Prof. Dr. Gökhan ERDEMİR

June, 2025 - 110 +XV Pages

This study aims to transmit real-time image data in Internet of Things (IoT) environments securely and with preserved integrity via a blockchain-based system. The research focuses on the secure transfer of images obtained from IoT devices by employing the Hyperledger Besu platform together with smart contracts. Within this scope, dynamic and static images are captured using TurtleBot and Raspberry Pi hardware; the data are partitioned into 128 KB fragments for efficient transfer, and their integrity is verified with the SHA-256 hash algorithm. The system operates on a four-node Hyperledger Besu network configured with the QBFT consensus algorithm, thereby ensuring the trustworthy storage of all records. Experimental results confirm the system's effectiveness in maintaining data integrity and enabling privacy-oriented, real-time secure data transmission.

The thesis is organised into nine chapters: research problem, literature review, fundamentals of blockchain and IoT, methodology and implementation, system design, performance evaluation, security analysis, discussion, conclusions, and future work.

Keywords: Blockchain, Internet of Things, image data transmission, Hyperledger Besu, smart contracts, data integrity, consensus algorithms.

İÇİNDEKİLER

TABLOLAR

ŞEKİLLER

KISALTMALAR

ABE	Öznitelik Tabanlı Şifreleme	Attribute-Based Encryption
ABFT	Asenkron Bizans Hata Toleransı	Asynchronous Byzantine Fault Tolerance
API	Uygulama Programlama Arayüzü	Application Programming Interface
Avalanche	Olasılıksal Mutabakat Protokolü	Probabilistic Consensus Protocol
BAN	Burrows–Abadi–Needham	Burrows–Abadi–Needham
BFT	Bizans Hata Toleransı	Byzantine Fault Tolerance
BIoT	Blok Zincir Nesnelerin İnterneti	Blockchain Internet of Things
CBC	Doğruya Göre İnşa Edilmiş	Correct-by-Construction
CASPER	Casper Protokolü	Casper Protocol
Chained BFT	Zincirlenmiş Blok Doğrulama	Sequential Block Validation
CRSM	Mutabakat Kaynak Dilimleme Modeli	Consensus Resource Slicing Model
DAG	Yönlendirilmiş Asiklik Grafik	Directed Acyclic Graph
DAG-Based	Yönlendirilmiş Asiklik Grafik Tabanlı Mutabakat	DAG-Based Consensus
DLT	Dağıtık Defter Teknolojisi	Distributed Ledger Technology
DPoW	Deterministik İş Kanıtı	Deterministic Proof of Work
dPoS	Yetkilendirilmiş Hisse Kanıtı	Delegated Proof of Stake
EVM	Ethereum Sanal Makinesi	Ethereum Virtual Machine
FFG	Dostane Sonuçlandırma Aracı	Friendly Finality Gadget
HDPoR	Hiper Delege Rastgelelik Kanıtı	Hyper Delegated Proof of Randomness
HotStuff	Doğrusal BFT Mutabakat Algoritması	Linear BFT Consensus Algorithm
HQ	BFT için Hibrit Quorum Protokolü	Hybrid Quorum Protocol for BFT
HTTPS	Güvenli Hiper Metin Aktarma Protokolü	HyperText Transfer Protocol Secure
HTTP	Hiper Metin Aktarma Protokolü	HyperText Transfer Protocol
IBFT	Istanbul BFT	Istanbul Byzantine Fault Tolerance
IoT	Nesnelerin İnterneti	Internet of Things
IPFS	Gezegenler Arası Dosya Sistemi	InterPlanetary File System
LEAP	Lider Seçim Algoritması Protokolü	Leader Election Algorithm Protocol
PBFT	Uygulamalı Bizans Hata Toleransı	Practical Byzantine Fault Tolerance
P2P	Eşler Arası İletişim	Peer-to-Peer
PoA	Otorite Kanıtı	Proof of Authority
PoActivity	Etkinlik Kanıtı	Proof of Activity
PoB	Yakım Kanıtı	Proof of Burn
PoC	Kapasite Kanıtı	Proof of Capacity
PoD	Mevduat Kanıtı	Proof of Deposit
PoET	Geçen Zaman Kanıtı	Proof of Elapsed Time
PoElapsed	Geçen Zaman Kanıtı	Proof of Elapsed Time

PoHistory	Tarihçe Kanıtı	Proof of History
PoI	Önem Kanıtı	Proof of Importance
PoLuck	Şans Kanıtı	Proof of Luck
PoRx	İtibar Kanıtı	Proof of Reputation
PoS	Hisse Kanıtı	Proof of Stake
PoSpace	Alan Kanıtı	Proof of Space
PoStorage	Depolama Kanıtı	Proof of Storage
PoW	İş Kanıtı	Proof of Work
PoWeight	Ağırlık Kanıtı	Proof of Weight
PoX	Transfer Kanıtı	Proof of Transfer (or Exchange)
RAFT	Çoğaltılmış Durum Makinesi	Replicated State Machine
RBFT	Yedekli Bizans Hata Toleransı	Redundant Byzantine Fault Tolerance
RPC	Uzak Prosedür Çağrısı	Remote Procedure Call
RPCA	Ripple Protokolü Mutabakat Algoritması	Ripple Protocol Consensus Algorithm
SCP	Stellar Mutabakat Protokolü	Stellar Consensus Protocol
SHA-256	256-Bit Güvenli Özetleme Algoritması	Secure Hash Algorithm 256-bit
Snowflake	Avalanche Ailesi Protokolleri	Avalanche Family Protocols
SPoF	Tek Bir Noktadan Kaynaklanabilecek Hata	Single Point of Failure
SSL	Güvenli Yuva Katmanı	Secure Sockets Layer
TPS	Saniye Başına İşlem	Transactions Per Second
TLS	Taşıma Katmanı Güvenliği	Transport Layer Security
VRF	Doğrulanabilir Rastgele Fonksiyon	Verifiable Random Function
ZK-SNARK	Sıkıştırılmış Etkileşimsiz Bilgi Argümanı	Succinct Non-Interactive Argument of Knowledge
ZKP	Sıfır Bilgi İspatı	Zero-Knowledge Proof
ABE	Öznitelik Tabanlı Şifreleme	Attribute-Based Encryption
ABFT	Asenkron Bizans Hata Toleransı	Asynchronous Byzantine Fault Tolerance
API	Uygulama Programlama Arayüzü	Application Programming Interface
Avalanche	Olasılıksal Mutabakat Protokolü	Probabilistic Consensus Protocol
BAN	Burrows–Abadi–Needham	Burrows–Abadi–Needham
BFT	Bizans Hata Toleransı	Byzantine Fault Tolerance
BIoT	Blok Zincir Nesnelerin İnterneti	Blockchain Internet of Things
CBC	Doğruya Göre İnşa Edilmiş	Correct-by-Construction
CASPER	Casper Protokolü	Casper Protocol
Chained BFT	Zincirlenmiş Blok Doğrulama	Sequential Block Validation

CRSM	Mutabakat Kaynak Dilimleme Modeli	Consensus Resource Slicing Model
DAG	Yönlendirilmiş Asiklik Grafik	Directed Acyclic Graph
DLT	Dağıtık Defter Teknolojisi	Distributed Ledger Technology
DPoW	Deterministik İş Kanıtı	Deterministic Proof of Work
dPoS	Yetkilendirilmiş Hisse Kanıtı	Delegated Proof of Stake
EVM	Ethereum Sanal Makinesi	Ethereum Virtual Machine
FFG	Dostane Sonuçlandırma Aracı	Friendly Finality Gadget
HDPoR	Hiper Delege Rastgelelik Kanıtı	Hyper Delegated Proof of Randomness
HotStuff	Doğrusal BFT Mutabakat Algoritması	Linear BFT Consensus Algorithm
HQ	BFT için Hibrit Quorum Protokolü	Hybrid Quorum Protocol for BFT
HTTPS	Güvenli Hiper Metin Aktarma Protokolü	HyperText Transfer Protocol Secure
HTTP	Hiper Metin Aktarma Protokolü	HyperText Transfer Protocol
IBFT	Istanbul BFT	Istanbul Byzantine Fault Tolerance
IoT	Nesnelerin İnterneti	Internet of Things
IPFS	Gezegenler Arası Dosya Sistemi	InterPlanetary File System
LEAP	Lider Seçim Algoritması Protokolü	Leader Election Algorithm Protocol
PBFT	Uygulamalı Bizans Hata Toleransı	Practical Byzantine Fault Tolerance
P2P	Eşler Arası İletişim	Peer-to-Peer
PoA	Otorite Kanıtı	Proof of Authority
PoActivity	Etkinlik Kanıtı	Proof of Activity
PoB	Yakım Kanıtı	Proof of Burn
PoC	Kapasite Kanıtı	Proof of Capacity
PoD	Mevduat Kanıtı	Proof of Deposit
PoET	Geçen Zaman Kanıtı	Proof of Elapsed Time
PoHistory	Tarihçe Kanıtı	Proof of History
PoI	Önem Kanıtı	Proof of Importance
PoLuck	Şans Kanıtı	Proof of Luck
PoRx	İtibar Kanıtı	Proof of Reputation
PoS	Hisse Kanıtı	Proof of Stake
PoSpace	Alan Kanıtı	Proof of Space
PoStorage	Depolama Kanıtı	Proof of Storage
PoW	İş Kanıtı	Proof of Work
PoWeight	Ağırlık Kanıtı	Proof of Weight
PoX	Transfer Kanıtı	Proof of Transfer (or Exchange)

RAFT	Çoğaltılmış Durum Makinesi	Replicated State Machine
RBFT	Yedekli Bizans Hata Toleransı	Redundant Byzantine Fault Tolerance
RPC	Uzak Prosedür Çağrısı	Remote Procedure Call
RPCA	Ripple Protokolü Mutabakat Algoritması	Ripple Protocol Consensus Algorithm
SCP	Stellar Mutabakat Protokolü	Stellar Consensus Protocol
SHA-256	256-Bit Güvenli Özetleme Algoritması	Secure Hash Algorithm 256-bit
Snowflake	Avalanche Ailesi Protokolleri	Avalanche Family Protocols
SPoF	Tek Bir Noktadan Kaynaklanabilecek Hata	Single Point of Failure
SSL	Güvenli Yuva Katmanı	Secure Sockets Layer
TPS	Saniye Başına İşlem	Transactions Per Second
TLS	Taşıma Katmanı Güvenliği	Transport Layer Security
VRF	Doğrulanabilir Rastgele Fonksiyon	Verifiable Random Function
ZK-SNARK	Sıkıştırılmış Etkileşimsiz Bilgi Argümanı	Succinct Non-Interactive Argument of Knowledge
ZKP	Sıfır Bilgi İspatı	Zero-Knowledge Proof

BİRİNCİ BÖLÜM

GİRİŞ

Son yıllarda, Nesnelerin İnterneti (IoT-Internet of Things) teknolojisinin çeşitli alanlarda hızla yaygınlaşmasıyla birlikte, özellikle görsel verilerin (resim, video ve canlı görüntü aktarımı gibi) güvenli şekilde iletilmesi ve depolanması önemli bir sorun haline gelmiştir. Bu veriler gizlilik ve güvenlik bakımından hassas bilgileri içermesi nedeniyle daha yüksek güvenlik standartları gerektirmektedir. Ancak mevcut IoT altyapılarının çoğu, merkezi ve bulut tabanlı sistemlerde tutulmakta olup, bu durum gizlilik ihlalleri, güvenlik açıkları ve veri manipülasyonlarına yol açabilmektedir (Liu et al. 2022), (Hu et al. 2022), (Tran, Ali Babar, and Boan 2021).

Bulut tabanlı IoT sistemlerinin karşılaştığı temel sorunlardan biri de merkezi yapılarda tek bir noktadan kaynaklanabilecek hata (single point of failure) riskidir (Liu et al. 2022), (Tran et al. 2021). Ayrıca, geleneksel merkezi sistemlerde verilerin bütünlüğünü ve mahremiyetini garanti altına almak her zaman mümkün olmayabilir.

Bu sorunlara çözüm olarak blok zincir teknolojisi, merkezi olmayan yapısı, güçlü kriptografik algoritmaları ve akıllı sözleşmeler yoluyla veri akışını otomatikleştirme yeteneği sayesinde IoT uygulamalarında güvenli bir alternatif haline gelmiştir (Liu et al. 2024), (Hu et al. 2022), (Commey et al. 2024). Blok zincirin dağıtık defter yapısı, verilerin güvenli, doğrulanabilir ve değiştirilemez şekilde depolanmasını sağlamaktadır (Liu et al. 2024), (Martina et al. 2023).

Blok zincir teknolojisinin ilk kullanımı, blok zincir teknolojisine dayanan ve çevrimiçi olarak mal alışverişi yapmak için geleneksel para değişimlerine benzer bir yöntem kullanan dijital bir para birimi olan Bitcoin'dir. Bitcoin'in popülaritesi sayesinde, insanlar artık finansal piyasa, IoT, tedarik zinciri, oylama, tıbbi tedavi ve depolama gibi geniş bir alan ve hizmet yelpazesinde blok zinciri teknolojilerini kullanabilirler. 2008'de blok zincirinin ilk merkeziyetsiz kripto para biriminin temeli olarak tanıtılması, güvenli, etkili ve şeffaf bir eşler arası bilgi alışverişine olanak sağladı. Blok zincir, işlemleri kronolojik sırayla kaydeden ve uygun bir mutabakat yöntemiyle güvence altına alınan, herkese açık, değiştirilemez bir defterdir (Puthal et al. 2018). Öne çıkan özellikleri arasında değişmezlik, geri döndürülemezlik, merkeziyetsizlik,

kalıcılık ve anonimlik yer alır. Blok zinciri, verileri dağıtık bir şekilde depolayarak sürekli artan kayıtları koruma altına alan bir sistemdir. Programları ve işlemleri içeren bloklardan oluşur ve her işlem ağın tamamında görülebilir. Bu teknoloji, güvenlik, gizlilik ve verimliliği artırarak veri manipülasyonu ve yetkisiz erişim gibi risklere karşı koruma sağlar. Her düğümün tüm blok zincirin bir kopyasını içerdiği, veri bütünlüğünü ve değiştirilemezliğini garanti eden eşler arası bir ağ üzerinde çalışır (Patel et al. 2020a).

Blok zinciri, iş ve endüstri dünyası için yeni ve gelişmiş özellikler sunarken aynı zamanda çeşitli endüstrilerdeki sorunları çözme yetenekleri nedeniyle son zamanlarda büyük ilgi görmektedir. Bu teknolojiler, güvenli veri kaydı ve paylaşımı, tedarik zinciri süreçlerinin otomasyonu ve şeffaflığın artırılması gibi sorunları ele almak için etkili bir yol sunar. Blok zincirinin kullanımı, verimlilik ve güvenliği artırırken, izlenebilirlik ve şeffaflığı geliştirir ve maliyetleri düşürerek birçok endüstriyel uygulamayı mümkün kılar (Al-Jaroodi and Mohamed 2019).

Blok zinciri teknolojisi, eşler arası ağların ve kriptografik algoritmaların avantajlarını birleştirerek anlaşmaların geçerliliğini garanti eder. Tüm katılımcı varlıklar, diğer katılımcı varlıkları dahil etmeden onaylanmış ve kaydedilmiş bir faaliyeti değiştiremez. Bu özellik, bir grup varlık arasında çeşitli iş anlaşmaları yürütmek için çok uygundur. Blok zinciri ayrıca olayların sırasını koruyabilir ve zaman içinde kaydedilen işlemlerin doğruluğunu garanti edebilir. Hiç kimse kaydedilen işlemlerden herhangi birini tek başına değiştiremeyeceği için kayıtları sahtelemek veya bir anlaşmadan vazgeçmek neredeyse imkansızdır. Sonuç olarak, çeşitli sektörlerde ve işletmelerde blok zinciri kullanımını değerlendirmek ve bunların bu sektörlerde etkili bir şekilde uygulanabilmesi için daha fazla araştırma yapılmaktadır (Al-Jaroodi and Mohamed 2019).

İzinsiz, izinli, hibrit ve konsorsiyum blok zincir teknolojisi çeşitli kullanım senaryolarına göre kategorize edilmiştir. Blok zincirlerin her biri farklı ihtiyaç ve zorluklara yanıt vermek için tasarlanmıştır, ancak belirli durumlar için hangi blok zincirin daha iyi olduğu konusunda hala tartışmalar var. İzinsiz blok zincirler (Bitcoin ve Ethereum gibi), tamamen merkeziyetsizdir ve herkesin katılımına açıktır. Bununla birlikte, bu yapıların ölçeklenebilirliği, işlem hızları ve maliyetleri gibi sorunları olabilir. Hyperledger Fabric gibi izinli blok zincirler daha merkeziyetçi ve kontrollü

bir yapıya sahiptir. Bu sistemlerde işlem yapmak için yalnızca yetkilendirilmiş katılımcılar gereklidir, bu da güvenlik ve gizlilik açısından avantajları sağlarken merkeziyetsizliğin bazı avantajlarını da ortadan kaldırır.

İzinli blok zincirlerin, izinsiz blok zincirlerin sorunlarını çözme yeteneğine sahip olup olmadığına dair çeşitli görüşler var (Solat, Calvez, and Naït-Abdesselam 2021). İzinli blok zincirlerin, izinsiz blok zincirlerin şu anda karşılaştığı sorunlara (örneğin, işlem hızları, ağ ölçeklenebilirliği ve maliyet gibi) yeterli bir çözüm sunmadığını savunan bazı araştırmacılar var. İzinsiz blok zincirlerin sunduğu tam merkeziyetsiz yapı nedeniyle, bazı uzmanlar izinli blok zincirlerin güven ve maliyet açısından izinsiz blok zincirlerle rekabet edemeyeceğini öne sürmektedirler (Solat, Calvez, and Naït-Abdesselam 2021).

Bununla birlikte, hibrit blok zincirler, izinli ve izinsiz yapıların avantajlarını bir araya getirerek özel ve kamuya açık işlemleri dengelemeye yardımcı olur. Örneğin, bir şirket izinli bir blok zincir kullanarak tedarik zinciri işlemlerini yürütebilir, ancak izinsiz bir blok zincir kullanarak finansal işlemlerini yönetebilir. Bu, şeffaflık ve güvenlik sağlarken aynı zamanda performans ve güvenlik sağlar. Konsorsiyum blok zincirleri, birden fazla kuruluşun birlikte çalışarak blok zinciri kontrol ettiği yarı özel bir sistemdir. Bu model, bankalar veya büyük tedarik zincirleri gibi bir grup iş ortağının birlikte çalıştığı durumlarda faydalıdır çünkü izinli yapının güvenliği ve çok taraflı iş birliğinin esnekliği birleştirir. Aynı zamanda konsorsiyum blok zincirleri, birden fazla paydaşın ihtiyaçlarını dengeledikleri için (Liu et al. 2022), (Hu et al. 2022), (Tran et al. 2021).

Bulut tabanlı IoT sistemlerinin karşılaştığı temel sorunlardan biri de merkezi yapılarda SPoF riskidir ve sürdürülebilirliği için, blok zincir teknolojisi değerlendirmesi hem teknik hem de sürdürülebilir nitelikleri içermeli ve karar alma süreçleri psikolojik faktörleri ve farklı görüşleri hesaba katmalıdır (Bai and Sarkis 2020). Her birinin güçlü ve zayıf yönleri olan blok zincir türlerinin çeşitliliği, en uygun çözümü seçerken proje gereksinimlerinin dikkatlice değerlendirilmesini gerektirir (Ismail and Materwala 2019), (Farshidi et al. 2020).

Son dönemde yapılan çalışmalar, IoT sistemlerinde blok zincir kullanımının yaygınlaştığını ve güvenlik avantajları sunduğunu ortaya koymaktadır (Liu et al.

2022), (Lu et al. 2020), (Tran et al. 2021), (Commey et al. 2024). Ancak bu entegrasyon ölçeklenebilirlik, performans kısıtları, enerji tüketimi ve veri gizliliği gibi zorlukları da beraberinde getirmektedir (Tran et al. 2021), (Commey et al. 2024).

Bu çalışmanın temel amacı, blok zincir teknolojisi kullanılarak IoT sistemlerinde görüntü aktarımının güvenli ve etkili bir şekilde gerçekleştirilmesini sağlayacak bir sistem modeli gelistirmektir. Bu kapsamda, Hyperledger Besu tabanlı 4 düğümlü yerel bir blok zincir ağına dayalı, akıllı sözleşmeler ve gelişmiş kriptografik yöntemlerle desteklenen yenilikçi bir IoT görüntü aktarım sistemi sunulmuştur. Önerilen sistemde, TurtleBot ve Raspberry Pi cihazlarından alınan görüntülerin, HTTP RPC protokolü üzerinden Hyperledger Besu blok zincir ağına güvenli bir şekilde aktarımı sağlanmıştır. Veriler, blok zincire eklenmeden önce özetleme algoritması ile doğrulanmış ve büyük boyutlu veriler parçalama (chunking) yöntemiyle optimize edilmiştir. Bu yöntem, veri bütünlüğünü korurken, IoT cihazlarından gelen verilerin güvenli şekilde saklanmasını sağlar. Sistem kontrolü, PC üzerinden sağlanmakta olup performans analizleri ve güvenlik değerlendirmeleri gerçekleştirilerek önerilen modelin, literatürdeki mevcut çalışmalarla karşılaştırmalı olarak üstün yönleri ortaya konulmuştur.

Geliştirilen sistemin genel işleyiş akışı Şekil 1.1'de sunulmaktadır. IoT cihazları tarafından toplanan görüntü verileri, parçalanarak işlenmekte; bütünlük doğrulaması yapıldıktan sonra Hyperledger Besu tabanlı blok zincir ağına kaydedilmekte ve yalnızca yetkilendirilmiş kullanıcıların erişimine açılmaktadır.

Şekil 1. 1: Önerilen sisteminin genel işleyiş diyagramı.

Tezin ilerleyen bölümleri aşağıdaki şekilde yapılandırılmıştır: Bölüm II'de ilgili literatür çalışmaları kapsamlı şekilde incelenecek, Bölüm III'te blok zincir ve IoT teknolojilerinin temel kavramları tanıtılacak, Bölüm IV'te yöntem ve uygulama adımlarına, Bölüm V' te önerilen sistemin detayları verilecek, Bölüm VI'da sistem

performans değerlendirmeleri yer alacak, Bölüm VII' de güvenlik analizleri sunulacak, Bölüm VIII' de tartışma ve çalışmanın literatüre katkısına yer verilirken son olarak Bölüm IX' da çalışma özetlenerek gelecekte yapılabilecek çalışmalar önerilecektir.

PROBLEM TANIMI

IoT sistemlerinde sayısız cihaz, özellikle kamera ve sensörler aracılığıyla büyük miktarda görsel veri (resim, video, canlı akış) üretip iletmektedir. Bu görsel verinin iletimi ve depolanması sürecinde güvenlik ve gizlilik kritik sorunlar olarak ortaya çıkmaktadır; örneğin, video tabanlı IoT verileri yetkisiz erişime karşı savunmasız olup, içerdikleri hassas bilgiler nedeniyle gizlilik ihlalleri riski yüksektir (Moolikagedara et al. 2024). Ayrıca, yüksek boyutlu görüntü ve video verilerinin gerçek zamanlı işlenmesi ve taşınması mevcut altyapılarda ciddi performans kısıtlarına yol açmaktadır (Moolikagedara et al. 2024). Bu durum, IoT ortamlarında veri bütünlüğünü, mahremiyeti ve sistem verimliliğini korumayı güçleştirmektedir.

Birçok IoT dağıtımı, günümüzde veriyi iletmek ve saklamak için bulut tabanlı merkezi mimarilere dayanmaktadır. Ancak bu merkezi yapılar, doğası gereği kritik zafiyetler barındırır. Merkezi sunucular sistemde SPoF oluşturarak herhangi bir arıza veya saldırıda tüm ağı etkileyebilecek bir kırılganlık yaratır (Alajlan, Alhumam, and Frikha 2023). Bunun yanı sıra, verinin tek bir elde toplanması olası veri ihlalleri ve yetkisiz manipülasyon riskini arttırmaktadır (Alajlan et al. 2023). Örneğin, merkezi bir bulut sunucusuna yetkisiz erişim gibi durumlarda, hassas görüntü kayıtları kötü niyetli kişilerce sızdırılabilir veya değiştirilebilir. Bu gibi sorunlar, mevcut bulut tabanlı IoT çözümlerinde güven ve mahremiyetin sağlanmasını zorlaştırmaktadır.

Bu sınırlamalara bir yanıt olarak, blok zincir teknolojisi IoT veri yönetimi için umut vadeden bir çözüm olarak ortaya çıkmıştır. Blok zincir, dağıtık defter yapısıyla veriyi ağdaki pek çok düğümde kopyalayarak tek bir arıza noktasını ortadan kaldırır ve böylece sistemin dayanıklılığını artırır. Bunun yanında, blok zincirin değiştirilemezlik özelliği sayesinde bir kez kaydedilen veriler sonradan silinemez veya değiştirilemez; bu da IoT cihazlarından toplanan verilerin sonradan kurcalanmasını önler. Kriptografik mekanizmalar (örn. özet fonksiyonları ve dijital imzalar) ile blok zincir üzerindeki kayıtların bütünlüğü ve kaynağı güvence altına alınmakta, böylece verilerin

doğruluğu ve gizliliği korunmaktadır. Ayrıca, akıllı sözleşmeler teknolojisi sayesinde IoT cihazları arasında önceden tanımlanmış koşullara bağlı olarak otomatik ve güvenilir işlemler gerçekleştirilebilir (Baird, Harmon, and Madsen 2018). Bu özellikler, blok zinciri tabanlı IoT sistemlerinin güvenlik ve emniyet açısından mevcut merkezi modellere kıyasla önemli avantajlar sunabileceğini göstermektedir.

Öte yandan, blok zincirin IoT ekosistemine entegrasyonu da kendi zorluklarını beraberinde getirmektedir. Özellikle ölçeklenebilirlik ve enerji tüketimi temel engeller olarak karşımıza çıkmaktadır; blok zincir ağları yoğun IoT veri trafiğinde darboğaza girerek işlemlerde gecikmeye ve düşük işlem hacmine yol açabilmektedir (Haque et al. 2024). Ayrıca, bazı blok zincir mutabakat mekanizmaları (ör. PoW) yüksek enerji tüketimi gerektirir; pil ile çalışan IoT cihazları için bu sürdürülemez bir durumdur. Üstelik IoT cihazlarının kısıtlı işlemci ve bellek kaynakları, bu cihazların tam teşekküllü bir blok zincir düğümü olarak çalışmasını pratik olmaktan çıkarmaktadır (Baird et al. 2018). Dolayısıyla, blok zincir tabanlı bir IoT çözümü geliştirirken ölçeklenebilirlik, enerji verimliliği ve düşük gecikme gereksinimleri gibi konular kritik önem arz etmektedir.

Bu tez çalışması, yukarıda belirtilen boşluğu doldurmayı hedefleyerek IoT cihazlarından elde edilen görsel verilerin güvenli, bütünlüğü korunmuş ve verimli bir şekilde blok zincir tabanlı bir sisteme aktarılması problemini ele almaktadır. Amaç, bir IoT kamerası tarafından üretilen bir görselin uçtan uca güvenli biçimde blok zincir ağına aktarılmasını ve kaydedilmesini sağlamaktır. Bu kapsamda, verinin bütünlüğünün garanti altına alınması için her bir görüntüye ait kriptografik özetleme değerleri hesaplanarak blok zincirine kaydedilecek; böylece herhangi bir değişiklik girişimi anında tespit edilebilecektir. Öte yandan, yüksek boyutlu görsel verilerin verimli depolanabilmesi ve iletim gecikmesinin azaltılabilmesi için parçalama yöntemleri kullanılacaktır. Bu sayede, görüntü verisi uygun boyutlu parçalara ayrılarak dağıtık ağda depolanacak ve blok zinciri üzerinde yalnızca referans özetleme değerleri tutulacaktır (Xu, Ren, and Zhu 2023). Önerilen yaklaşım, IoT tabanlı görüntü iletiminde blok zincir teknolojisi kullanılarak güvenlik ve veri bütünlüğü sorununa çözüm getirmeyi ve mevcut merkezi yapıya duyulan ihtiyacı ortadan kaldırmayı amaçlamaktadır.

İKİNCİ BÖLÜM

LİTERATÜR İNCELEMESİ

Blok zincir ve IoT teknolojilerinin entegrasyonu üzerine son yıllarda yapılan birçok çalışma, bu iki teknolojinin birbirlerini tamamlayıcı özellikleri nedeniyle önemli avantajlar sunduğunu göstermektedir. Bu bölümde, özellikle blok zincir tabanlı IoT sistemlerinde veri paylaşımı, güvenlik, gizlilik, akıllı sözleşmeler ve performans optimizasyonları üzerine odaklanmış literatür çalışmaları detaylı şekilde ele alınmaktadır.

2.1. Blok zincir tabanlı iot sistemleri

(Leng et al. 2022), çalışmasında, Hyperledger tabanlı bir model önerilmiştir. Çalışmada, IoT cihazlarının daha güvenli ve verimli çalışmasını sağlamak için düşük enerji tüketen Mutabakat algoritmaları, akıllı işlem doğrulama yöntemleri ve zincir içi (on-chain), zincir dışı (off-chain) veri depolama stratejileri kullanılmıştır. Bu yaklaşım, akıllı şehirler, enerji yönetimi ve ulaşım gibi alanlarda IoT cihazlarının performansını artırmak için etkili bir yöntem olarak öne çıkmaktadır.

IoT sistemlerinde veri güvenliği önemli bir sorundur. (Y. Li et al. 2020), bu soruna çözüm olarak blok zincir tabanlı güvenli görüntü aktarım modeli geliştirmiştir. Çalışmada, IoT cihazlarından alınan görüntü verileri küçük parçalara ayrılarak her parçaya benzersiz bir imza eklenmiştir. Bu yöntem, veri manipülasyonu, sahtecilik ve DoS saldırılarına karşı koruma sağlamaktadır. Ayrıca, modelin anahtar yönetimi ve imza doğrulama mekanizmalarıyla veri bütünlüğünün korunduğu belirtilmiştir. Bu çalışma, IoT sistemlerinde güvenli veri aktarımı için etkili bir çözüm sunmaktadır.

Liu ve arkadaşları (Liu et al. 2022), sıfır güven yaklaşımı kullanarak blok zincir tabanlı IoT sistemlerinde merkezi olmayan, adil ve doğrulanabilir bilgi paylaşım modeli önermektedir. Bu çalışma, akıllı sözleşmeler aracılığıyla katılımcıların kimlik doğrulamasını sağlamakta ve bilgilerin doğruluğunu garanti altına almaktadır. Sistem, Ethereum platformunda test edilmiş olup, düşük gecikme süreleri ile IoT sistemlerine uygun performans göstermiştir.

Liu ve ekibi (Liu et al. 2024), farklı blok zincir platformları üzerindeki akıllı sözleşme yürütme mekanizmalarını karşılaştırmalı olarak analiz etmiş ve akıllı sözleşme performansını artırmak için paralel yürütme, zincir dışı ve zincirler arası çözümler gibi çeşitli optimizasyon yöntemlerini incelemiştir. Bu çalışma, blok zincirin IoT sistemlerinde kullanılabilirliğini artıracak teknik iyileştirmeleri ortaya koymaktadır.

Lu ve arkadaşları (Lu et al. 2020), blok zincir ve birleşik öğrenme (federated learning) yöntemlerini bir araya getirerek, endüstriyel IoT (IIoT) sistemlerinde mahremiyet korumalı veri paylaşımını sağlamaktadır. Bu çalışma özellikle, farklılaştırılmış gizlilik ve eğitim kalitesinin kanıtı adlı yeni bir Mutabakat mekanizması sunarak, veri sağlayıcıların gizlilik endişelerini gidermeyi amaçlamaktadır.

Benzer şekilde, Hu ve arkadaşları (Hu et al. 2022) tüketici IoT uygulamaları için blok zincir destekli veri paylaşım şeması önermiştir. Önerilen sistemde, ABE ile veri erişimi güvence altına alınmakta, ayrıca aranabilir şifreleme algoritmalarıyla veri gizliliği artırılmaktadır. Sistemin performansı Hyperledger Fabric platformu üzerinde değerlendirilmiş ve IoT uygulamaları için tatmin edici sonuçlar elde edilmiştir.

Martina ve arkadaşları (Martina et al. 2023) ise Ethereum blok zincir üzerinde akıllı sözleşmelerin tasarımı, geliştirilmesi ve yürütülmesine dair genel bir bakış sağlamaktadır. Bu çalışma, akıllı sözleşmelerin yaşam döngüsü hakkında kapsamlı bilgiler vererek, IoT sistemlerinde bu sözleşmelerin etkin biçimde kullanımı için rehber niteliği taşımaktadır.

Tran ve arkadaşları (Tran et al. 2021), blok zincir ile Nesnelerin İnterneti (IoT) entegrasyonu üzerine kapsamlı bir sistematik literatür taraması sunmaktadır. Çalışmada, bu entegrasyonun temel motivasyonları olarak güvenlik, veri bütünlüğü ve merkeziyetsizlik öne çıkarılmaktadır. Ayrıca, blok zincir platformlarının uç–sis–bulut katmanlarında konumlandırılmasına yönelik tasarım kararlarının, IoT sistemlerinin genel başarısı üzerinde belirleyici bir rol oynadığı vurgulanmaktadır.

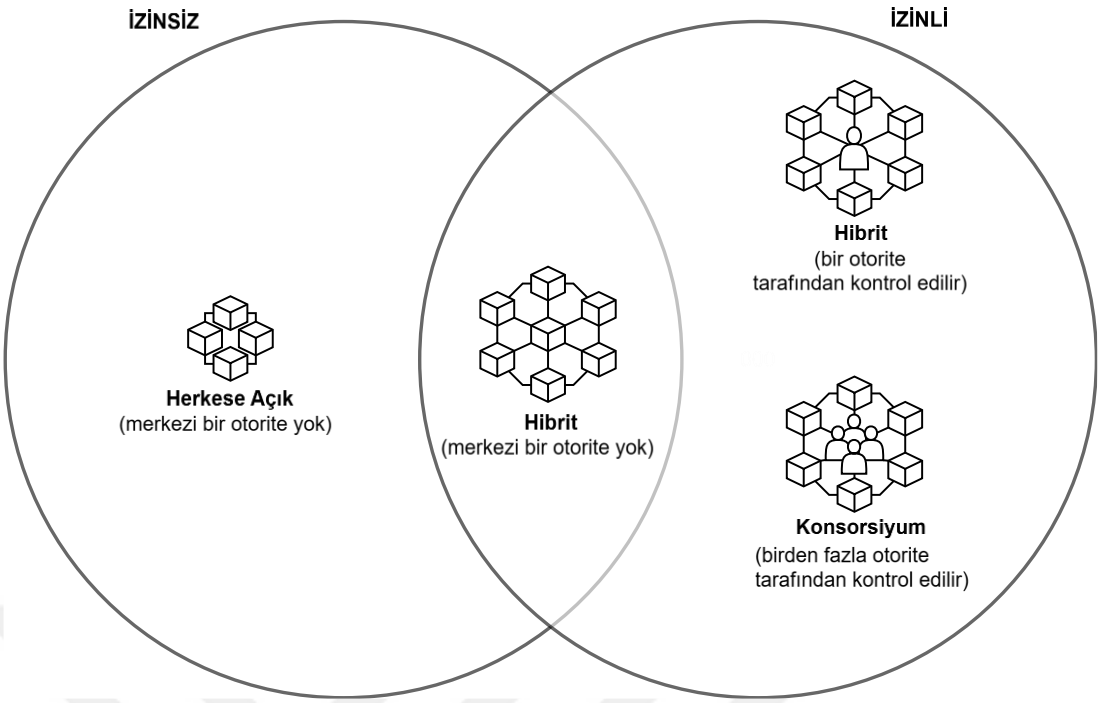
Skaria ve arkadaşları (Skaria, Bamini, and Chitra 2024), endüstriyel IoT ortamlarında blok zincir ve derin öğrenme modelleri (özellikle Bi-LSTM) kullanarak güvenli veri aktarımı için yenilikçi bir yöntem sunmaktadır. Önerilen modelde, sis ve uç bilişim

(fog ve edge computing) teknolojileri kullanılarak veri iletim gecikmesi azaltılmış ve tehditlere karşı gerçek zamanlı koruma sağlanmıştır.

Commey ve ekibi (Commey et al. 2024), blok zincir tabanlı IoT sistemlerinin güvenliği üzerine detaylı bir inceleme yapmış ve ZKP, PQC ve siber aldatma tekniklerinin kullanımının güvenliği önemli ölçüde artırdığını belirtmiştir. Ayrıca makine öğrenimi yöntemlerinin tehdit tespiti ve saldırı analizinde etkin olduğunu ortaya koymuştur.

(D. Li, Wong, and Guo 2020), (Wust and Gervais 2018), (Falazi et al. 2019), yaptıkları çalışmalarda blok zincirleri temelde iki türe ayırmakta ve bunları izinsiz ve izinli blok zincirler olarak sınıflandırmaktadır. Bu iki tür blok zincirin, farklı çalışmalarda genel ve özel blok zincirleri olarakta adlandırıldığı belirtilmiştir (Ferdous et al. 2020), (Falazi et al. 2019). Aslında iki ayrı adlandırılma olarak gözükse de birbirleri yerine kullanılan ifadelerin aslında aynı tür blok zincirleri temsil ettiğini görmekteyiz (Solat et al. 2021), (Helliar et al. 2020), (Falazi et al. 2019). Gizlilik, güvenlik gibi gereksinimlerin karşılanabilmesi için bu iki model, blok zinciri dünyasında büyük önem taşımaktadır (Kshetri 2017). Fakat Attaran ve arkadaşlarına göre dört tür blok zincir bulunmaktadır (Attaran and Gunasekaran 2019).

Daha detaylı literatür taraması yapıldığında bu farklı fikirlerin sebebinin aslında başlangıçta, blok zinciri modelleri izinsiz ve izinli olmak üzere iki temel kategoriye ayrılırken, teknolojik ilerlemeler ve sektörel ihtiyaçlar doğrultusunda bu sınıflandırma hibrit ve konsorsiyum blok zincirleri gibi üç ya da dört farklı türe genişlediği bilgisi anlaşılmıştır (Guru et al. 2023), (Santhoshi, Arigela, and Voola n.d.), (Chatziamanetoglou and Rantos 2024), (Verma, Jain, and Doriya 2021), (Yao et al. 2023a). Bu genişlemenin nedeni, teknoloji geliştikçe kullanım alanlarının, teknik detayların ve sistem gereksinimlerinin çeşitlenmesidir (Kim et al. 2022a), (Fraga-Lamas et al. 2024). Bu farklılığının sınıflandırıldığı sonucu Şekil 2.1(Guidi and Michienzi 2023) (Islam et al. 2023)' de görebilirsiniz.



Şekil 2. 1: Blok zincirlerin sınıflandırılması.

2.2. Görüntü verisi aktarımında blok zincir uygulamaları

(Durga et al. 2022) , Hyperledger Besu platformu kullanarak IoT cihazları için güvenli görüntü aktarımı sağlamak amacıyla akıllı sözleşme tabanlı bir sistem geliştirmiştir. Bu sistemde, veri bloklarının özetleme değerleri blok zincir üzerinde saklanarak görüntülerin bütünlüğü garanti altına alınmaktadır. Çalışmada, IoT cihazlarının sınırlı donanım kaynaklarına uygun hafifletilmiş kriptografik algoritmaların kullanılmasıyla gecikme süreleri optimize edilmiştir (Mohanta et al. 2021).

(Hasan et al. 2022; Xiong et al. 2020; Zhang et al. 2023), blok zincir temelli bir video akış sistemi tasarlayarak IoT ortamlarında gerçek zamanlı veri aktarımı sağlamıştır. Çalışmada, Ethereum platformunda geliştirilen bu modelin düşük gecikme süreleriyle çalıştığı ve veri kaybı yaşanmadığı rapor edilmiştir. Video akışının kesintisiz gerçekleşmesini sağlamak için özel durum kanalı protokollerinden yararlanılmıştır.

(S. et al. 2024), tarafından gerçekleştirilen bir çalışmada, Hyperledger Fabric platformunda gizli görüntü verilerinin güvenli depolanması ve aktarımı sağlanmıştır. Önerilen sistemde, ABE kullanılarak yalnızca yetkili kullanıcıların verilere erişebilmesi garanti edilmiştir (Rouhani et al. 2021) , (Liu et al. 2023) , (Sun et al. 2022). Ayrıca, sistemin performans analizlerinde CPU kullanımı ve işlem süresi

bakımından başarılı sonuçlar elde edilmiştir. Bu sistemler, güvenilir denetlenebilirlik ve şeffaflık sağlarken, anahtar emaneti ve dağıtımı gibi geleneksel ABE' deki temel yönetim sorunlarını ele alır (Rouhani et al. 2021), (Liu et al. 2023). Blok zinciri depolama baskısını hafifletmek için dağıtık merkeziyetsiz dosya sistem teknolojisiyle entegrasyon önerildi (Sun et al. 2022). Bu blok zinciri destekli ABE sistemleri, güvenli yükleme ve indirme aşamaları, kullanıcı kimlik doğrulaması ve verimli kullanıcı iptali sunarak, bunları bulut ortamlarında hassas verileri korumak için umut verici çözümler haline getiriyor (S. et al. 2024), (Liu et al. 2023).

(Vivek Anand and Vijayalakshmi 2020), IoT ortamlarında çeşitli konumlarda elde edilen görüntü verilerinin blok zincir üzerinde işlem (transaction) olarak saklandığı, güvenli ve doğrulanabilir bir görüntü aktarım modeli geliştirerek, bu yapının özellikle suçlu tespiti gibi siber istihbarat uygulamalarında etkin biçimde kullanılabileceğini ortaya koymuştur. Tablo 2.1' de literatürde yapılmış üç çalışma ile bu tezin tüm özgünlükleri karşılaştırılmıştır.

(Fitwi & Chen, 2021), izinli bir blokzincir altyapısı kullanarak CCTV gözetim videolarının güvenli şekilde paylaşılmasını amaçlayan bir sistem önermiştir. Bu yapıda, videolar zincir dışı saklanmakta, ancak bu videoların hash değerleri blokzincire yazılarak doğrulanabilirlik sağlanmaktadır. Sistem, temel düzeyde zaman damgası ve veri bütünlüğü kontrolü sunmakla birlikte, yalnızca video verisine odaklanmış ve çoklu veri türlerini desteklememektedir.

(Moolikagedara et al., 2023), araç üzeri kameralardan elde edilen video verilerinin merkeziyetsiz bir şekilde paylaşımı için blokzincir tabanlı bir sistem tasarlamıştır. Bu çalışmada, özellikle akıllı şehir senaryolarında dağıtık video görüntülerinin güvenli depolanması ve sürdürülebilirliği hedeflenmiştir. Ancak model, sadece video verisi ve dijital imzalar üzerinden işlem yaparak, veri bütünlüğü için sınırlı güvenlik önlemleri sunmuştur.

(Michelin et al., 2020), havaalanı gibi güven düzeyi değişken kurumların yer aldığı ortamlarda, gözetim kamerası videolarının bütünlüğünü sağlamak amacıyla hafif blokzincir teknolojisini temel alan bir yapı önermiştir. Geliştirilen sistemde, video dosyalarının kendisi değil, yalnızca metadata ve hash bilgileri blokzincire yazılarak doğrulama sağlanmakta; bu sayede hem veri bütünlüğü hem de reddedilemezlik (non-

repudiation) hedeflenmektedir. Ancak sistemin zaman/sıralama kontrolü zayıf kalmış ve yalnızca video veri türüyle sınırlandırılmıştır.

Tablo 2. 1: Literatür karşılaştırması.

Özellik / Çalışma	**Fitwi & Chen (2021)**	**Moolikagedara et al. (2023)**	**Michelin et al. (2019)**	**Ağgül, 2025**
Blok zincir Türü	Permissioned (özel zincir)	Ethereum benzeri blok zincir	Hafif blok zincir mimarisi	Hyperledger Besu
Veri Türü	CCTV video	Araç kameralarından alınan videolar	Gözetim görüntüleri (sabit kameralar)	Görüntü
Veri Aktarım Yapısı	Blokzincire hash + meta veri	Video parça hash'leri ve dijital imzalar	Zincir dışı veri + zincir içi bütünlük	Parçalanmış veri (base64) + hash + timestamp
Hash Kullanımı	SHA tabanlı	SHA + dijital imza	SHA-256	SHA-256 (her parça için)
Zaman Damgası / Sıralama	Sınırlı	Varlık bazlı	Belirtilmemiş	timestamp + chunk_id ile tam sıralama
Multimedya Desteği	Sadece video	Sadece video	Sadece görüntü	Görüntü + Ses + LIDAR + GPS
Blok zincire Yazılan	Hash + metadata	Hash + signature	Hash	Base64 veri parçası + hash + zaman
Özgünlük / Gelişmişlik	Görsel veri zincir dışında, zincir içi kısıtlı	Verinin tamamı zincire yazılmaz	Veri transferinde zayıf denetim	Tüm veri zincire bağlı, her parça ayrı ayrı doğrulanabilir

2.3. Akıllı sözleşmeler ve kriptografi yaklaşımları

(Khor et al. 2023), halka açık blok zincirlere bağlı IoT cihazları için işlem ücretlerini ve güç tüketimini azaltırken sensör verilerinin bütünlüğünü güvence altına alan bir protokol geliştirmiştir. (Chen, Wang, and Wang 2020), iş birliği yapan düğüm sayısını sınırlayan ve yükü uç düğümlere dağıtan stokastik bir blok zincir şeması sunarak güvenliği ve verimliliği artırmıştır.

Video bütünlüğünün doğrulanması için (Ghimire, Choi, and Lee 2020), özetleme tabanlı mesaj kimlik doğrulama ve eliptik eğri kriptografisini bir blok zincir çerçevesinde birleştiren bir yöntem önermiştir. (Lin et al. 2022) ise akıllı sözleşmeler aracılığıyla üçüncü taraf denetçilerin denetlenmesine olanak tanıyan, konsorsiyum blok zincir tabanlı bir sistem sunarak bulut depolamada kamuya açık bütünlük doğrulamasını mümkün kılmıştır.

Yapılan çalışmalar, IoT cihazlarında veri güvenliği ve blok zincir tabanlı doğrulama mekanizmalarına odaklanmış olsa da, IoT sistemlerinde görüntü verilerinin güvenli ve verimli aktarımı konusunda yeterince inceleme yapılmadığı gözlemlenmektedir. Bu çalışmada, bu eksikliği gidermek amacıyla IoT cihazlarından alınan görüntü verilerinin blok zincir tabanlı bir modelle güvenli bir şekilde aktarılması sağlanmıştır.

Geliştirilen modelde, IoT cihazlarının düşük güç tüketimi ve hızlı veri işleme gereksinimleri dikkate alınarak Python tabanlı optimize edilmiş bir akıllı sözleşme tasarlanmıştır. Bu model, görüntü verilerinin blok zincir üzerinde bütünlüğünü koruyarak güvenli aktarımını mümkün kılmaktadır. Sistem, Hyperledger Besu platformunda test edilmiş ve yüksek işlem doğruluğu ile düşük gecikme süreleri elde edilmiştir.

ÜÇÜNCÜ BÖLÜM

BLOK ZİNCİR VE IoT TEMELLERİ

Blok zincir teknolojisi, IoT sistemlerinde güvenlik ve gizlilik zorluklarını çözmek için umut verici çözümler sunmaktadır. Blok zincir ile IoT'nin entegrasyonu, BIoT olarak bilinir ve bu, merkeziyetsiz mutabakat değiştirilemezlik ve şeffaf işlem kayıtları sayesinde güvenliği, gizliliği ve veri bütünlüğünü artırır (Deepak et al. 2024), (Al-Shareeda, Saare, and Manickam 2023). Blok zincirin merkeziyetsizlik, mutabakat mekanizmaları ve akıllı sözleşmeler gibi özellikleri, potansiyel saldırıları önleyebilecek ve işlem maliyetlerini azaltabilecek dağıtık IoT sistemlerinin inşa edilmesi için oldukça uygundur (Xu, Lu, and Li 2021). Bu teknolojilerin birleşimi, akıllı sağlık, akıllı şebekeler ve akıllı şehirler gibi çeşitli alanlarda uygulamalara sahiptir (Abdelmaboud et al. 2022). Ancak, ölçeklenebilirlik, birlikte çalışabilirlik ve güvenilirlik gibi zorluklar devam etmektedir. Süregelen araştırmalar, IoT ortamları için özel olarak tasarlanmış blok zincir platformları geliştirmeye ve BIoT sistemlerinin potansiyelini tam olarak gerçekleştirebilmek için sınırlamaları aşmaya odaklanmaktadır (Abdelmaboud et al. 2022) , (Al-Shareeda et al. 2023).

3.1. Blok zincir teknolojisi

Blok zincir, merkezi olmayan bir yapıda, işlemleri güvenli ve doğrulanabilir şekilde gerçekleştiren, DLT teknolojisidir. Blok zinciri, eşler arası çalışan merkeziyetsiz bir platformdur (Deng, Huang, and Wang 2022). Mimarisi dağıtık olup, kaynaklarını her bir düğüme tahsis eder ve düğümler, ağ adına hangi taleplerin onaylanacağına birlikte karar verir (Deng et al. 2022). Merkeziyetsiz bir sistemde, tüm iletişimler için aracı veya temsilci olarak hizmet veren merkezi bir düğüm yoktur. Ancak tüm düğümler, işlemler olarak bilinen eşler arası işlevleri gerçekleştirme yetkisine sahiptir. Blok zinciri defteri, dağıtık olmasının yanı sıra sadece eklemeye açık bir yapıya sahiptir, yani bir işlem deftere girildikten ve doğrulandıktan sonra kaldırılması, değiştirilmesi veya üzerinde oynama yapılması mümkün değildir. Bu, işlemlerin her seferinde anında gerçekleşmesini sağlar (D. Li et al. 2020).

(Yuan and Wang 2018), blokzincir teknolojisinin yalnızca tek bir teknikle sınırlı kalmayıp; kriptografi, matematik, algoritmalar ve ekonomik modeller gibi çok çeşitli disiplinleri kapsayan çok yönlü bir yapı sunduğunu ifade etmektedir. Çalışmalarında,

dağıtılmış fikir birliği algoritmaları aracılığıyla eşler arası ağların entegre edildiğini ve bu sayede geleneksel dağıtık veritabanı senkronizasyon sorunlarının çözüldüğünü vurgulamaktadırlar (Gervais et al. 2014; Juan Garay 2015; Nakamoto n.d.). Bunun yanı sıra, blokzincirin altı katmanlı bir referans modeli üzerinden açıklanması, sistemin karmaşıklığını ve uygulama çeşitliliğini ortaya koymaktadır.

(De Filippi, Mannan, and Reijers 2020a) ise blokzincirin her ne kadar güvenilmez bir yapı olarak tanımlansa da, matematiksel kurallar, kriptografik ilkeler ve oyun teorisine dayalı teşvik mekanizmaları sayesinde bir "güven makinesi" olarak yeniden konumlandırılabileceğini savunmaktadır.

(Lin and Liao 2017), blok zincir teknolojisinin dağıtık, şeffaf, açık kaynaklı, bağımsız, değiştirilemez ve gizlilik sağlayan bir yapı sunduğunu ifade etmektedir. Bu teknoloji, verilerin merkezi bir otoriteye ihtiyaç duymaksızın dağıtık bir yapıda kaydedilmesi, depolanması ve güncellenmesini sağlayarak güvenli bir veri yönetimi sunar. Ağ üzerindeki tüm düğümlerin işlemleri görebilmesi ve güncellemelerin şeffaf bir şekilde gerçekleşmesi, sistemin denetlenebilirliğini artırır. Ayrıca blok zinciri sistemlerinin büyük bir kısmının açık kaynaklı olması, kullanıcıların bu altyapı üzerinde kendi uygulamalarını geliştirmelerine olanak tanır. Tek bir otoriteye bağlı olmadan çalışan blok zinciri, mutabakat mekanizmaları sayesinde tüm ağ üzerinde veri bütünlüğünü korur ve herhangi bir kaydın düğümlerin çoğunluğu ele geçirilmedikçe değiştirilememesini garanti eder. Son olarak, blok zinciri kullanıcılarının kimliklerini gizleyerek işlem yapabilmeleri, sistemin gizlilik ve anonimlik sağlama kapasitesini ortaya koymaktadır. Tüm bu özellikler, blok zinciri teknolojisini güvenilir, esnek ve çok yönlü bir çözüm olarak öne çıkarmaktadır.

Bu bağlamda, blok zincirin yalnızca genel nitelikleriyle değil, aynı zamanda alt bileşenleriyle de nasıl çalıştığını anlamak büyük önem taşımaktadır. Bu nedenle, bir sonraki alt bölümde blok zincirin temel bileşenleri olan veri yapısı, mutabakat algoritmaları, akıllı sözleşmeler ve blok zincir üzerinde güvenlik analizi detaylı bir şekilde ele alınacaktır.

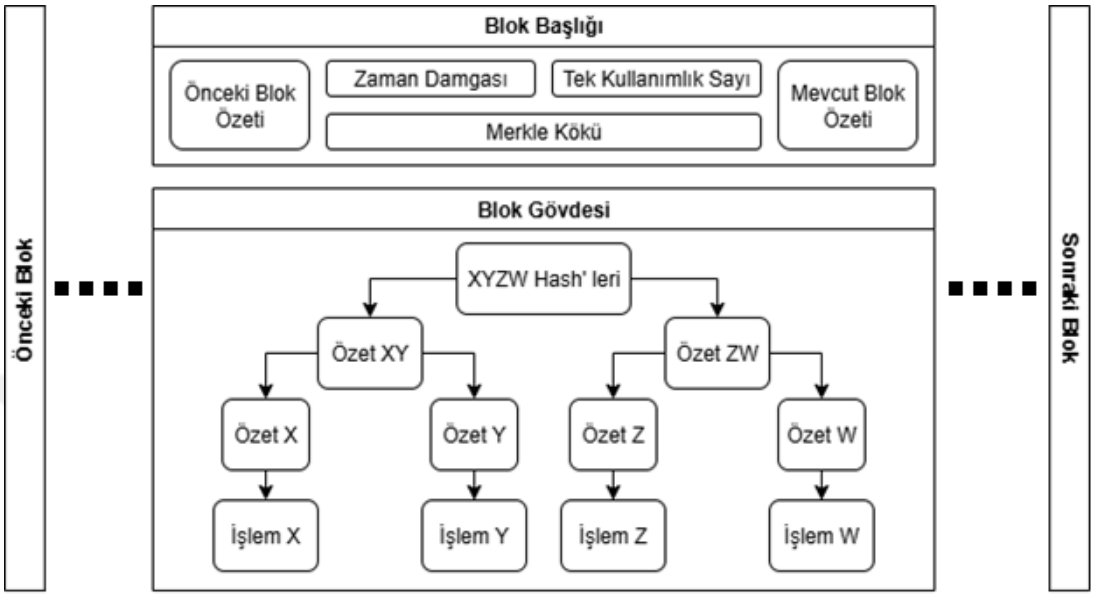
Blok zincir teknolojisinin oncusu olarak ilk olarak kripto para birimi olan Bitcoin oncelikli olarak finansal uygulamalarla literature girmistir. Ancak günümüzde IoT, sağlık, finans, lojistik gibi birçok farklı sektörde kullanım alanı bulmaktadır (Tran et al. 2021). Blok zincir teknolojisinin öne çıkan özellikleri şu şekilde sıralanabilir:

- Merkeziyetsizlik ve dağıtıklık, blok zincir, verileri merkezi bir otoriteye bağlı olmadan birçok düğüm(node) arasında dağıtır. Bu yapı, SPoF engelleyerek sistem güvenilirliğini artırır (Liu et al. 2022), (Tran et al. 2021).
- Blok yapısı ve kriptografi, veriler bloklar halinde kaydedilir ve her blok, kendinden önceki bloğa ait kriptografik özetleme değerini içerir. Bu sayede bloklar arasındaki bağlantı değiştirilmesi neredeyse imkânsız hale gelir ve veri bütünlüğü sağlanmış olur (Martina et al. 2023).
- Mutabakat algoritmaları, blok zincir ağında, blokların doğruluğunu ve geçerliliğini sağlamak amacıyla mutabakat algoritmaları kullanılır. PoW, PoS, PBFT gibi algoritmalar yaygın kullanılan mutabakat yöntemleridir (Liu et al. 2024), (Tran et al. 2021).
- Akıllı sözleşmeler (smart contracts), blok zincir üzerinde çalışan ve kendi kendini yürüten kod parçalarıdır. Akıllı sözleşmeler, işlemlerin otomatik, güvenilir ve şeffaf biçimde gerçekleşmesini sağlar. Özellikle Ethereum gibi platformlar üzerinde Solidity dili ile geliştirilen akıllı sözleşmeler, birçok IoT uygulamasında yoğun biçimde kullanılmaktadır (Liu et al. 2024), (Martina et al. 2023).
- Güvenlik ve gizlilik, blok zincir sistemleri, kriptografik yöntemler ve kimlik doğrulama mekanizmalarıyla güçlü bir güvenlik yapısı sunar. Ayrıca, ZKP gibi yöntemlerle kişisel verilerin gizliliği korunur (Commey et al. 2024).
- Ölçeklenebilirlik ve performans**,** blok zincir sistemlerinde, parçalama, yan zincir ve zincir dışı çözümler gibi tekniklerle performans artırılmakta ve IoT sistemleri gibi büyük hacimli veriler için ölçeklenebilirlik sağlanmaktadır (Liu et al. 2024), (Hu et al. 2022).

2.2. Blok zincir yapısı ve işlem akışı

IoT uygulamalarında, blok zincir teknolojisi işlem kayıtlarını bloklar halinde depolamak için kullanılan bir veri yapısı sunar (Mahajan 2019). Blok yapısı iki ana bileşenden oluşur: blok başlığı ve blok gövdesi (Gaba et al. 2022). Blok zinciri, zaman sıralı blokların bir araya gelmesiyle oluşan bir veri yapısıdır. Her blok, kayıtlar ve ilgili meta verileri içerir ve tüm işlem verilerini kaydeden bir zincir şeklinde birbirine bağlıdır (Patel et al. 2020b). Bu bağlılık blokları zincir benzeri bir yapı oluşturacak şekilde birleştirir. Bu nedenle, blok zinciri terimi kullanılır (Mahajan 2019). Aynı zamanda bir blok zinciri üzerindeki her blok için ana verilerin yanı sıra, önceki bloğun özetlemi, mevcut bloğun özetlemi, zaman damgasını içerir (Lin and Liao 2017). Bu

sayede her blok, benzersiz bir özetleme kimliğine sahip olur (Mahajan 2019). Şekil 3.1 (Mahajan 2019), (Guidi and Michienzi 2023), (Yao et al. 2023b), her bloğun sabit yapısını göstermektedir. Bu veri blokları, ağdaki tüm düğümlere eş zamanlı olarak ulaştırılmakta ve zincire sıralı şekilde eklenmektedir.

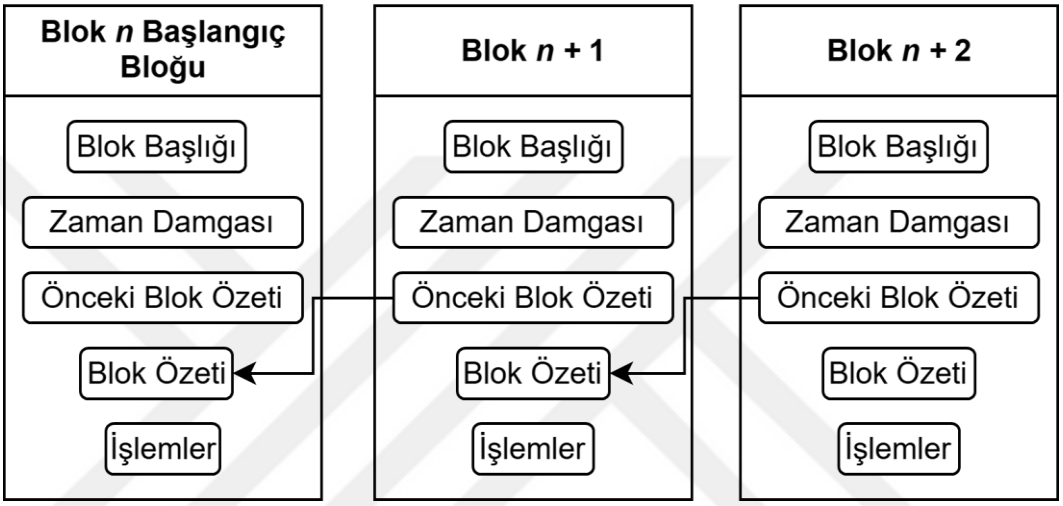


Şekil 3. 1: Standart blok zincir yapısı.

Blok gövdesi ve blok başlığı, Şekil 3.1'de gösterildiği gibi blokun iki ana bileşenidir. Bir bloğun başlığı, bloğun içeriğine ilişkin kısa bir özet sağlar. Önceki blok özetleminde ise önceki bloğun özetlemi bulunmaktadır. Bu, blokların birbirine zincirleme bağlanmasını sağlar ve böylece zincirin güvenliğini sağlar (Nurhidayat et al. 2021). Bu yapı Şekil 3.2(Nurhidayat et al. 2021)' de gösterilmektedir. Timestamp veya Zaman Damgası ise bloğun oluşturulduğu zamanı belirtir. Tek kullanımlık sayı olarak adlandırılan bu kısım PoW gibi algoritmalarda kullanılan özel bir değerdir. Blok zincirinin özetlem hesaplamasında kullanılır. Özetlem tipik olarak belirli gereksinimleri karşılamak için rastgele denemeler yaparak belirlenir. Mevcut bloğun özetlemi alanı ise bu bloğun tüm içeriğinin özetlemidir. Bu özetlem, içerikte herhangi bir değişiklik olursa tamamen değiştirilir. Blok işlemleri ise blok gövdesinde bulunur. Ayrıca Merkle kökü olarak adlandırılan bir veri yapısı, işlemleri organize eder. Merkle kökü özetlem fonksiyonları bloktaki tüm işlemleri düzenler. Bu ağaçta her işlem bir yaprak düğümünde gerçekleşir ve iki yaprağın özetleri birleştirilerek üst seviyede bir özetlem oluşturulur (Zheng et al. 2018). Tüm işlemleri özetleyen Merkel kökü oluşturuluncaya kadar bu işlem devam eder.

İşlem A, B, C ve D: Her bir işlem bir özetlem değeri ile gösterilir. Bu özetlem değerleri birleştirerek, üst seviyelerdeki özetler (AB, CD ve ABCD) oluşturulur.

Bu yapı, blok zincirindeki her bir bloğun verilerinin güvenliğini garanti eder. Herhangi bir işlem üzerinde oynama yapılırsa, değişiklik sadece ilgili işlemde değil, aynı zamanda bloğun özetleminde ve zincirin geri kalanındaki tüm bloklarda da görülecektir. Blok zincirinin şeffaflığı, güvenliği ve değişmezliği bu şekilde korunmaktadır.



Şekil 3. 2: Özetlem bağlantı yapısı.

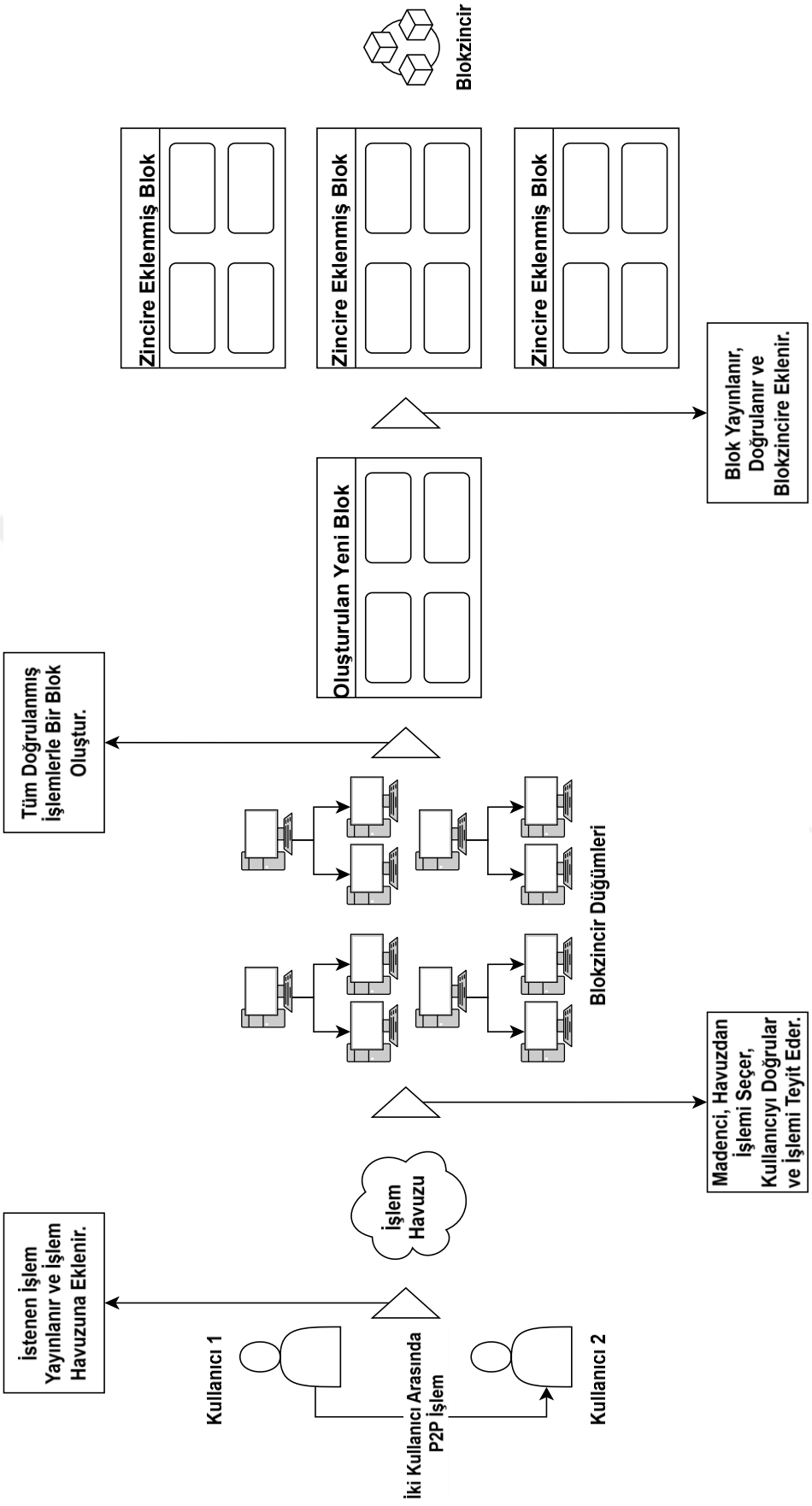
(Davidson, De Filippi, and Potts 2016) göre, her işlem bir imzaya sahiptir. Kullanıcılar bu imzayı kullanarak işlemlerini kriptografik olarak güvence altına alınmış bloklara dayanarak doğrular. Her kullanıcı, bir mutabakat sorununu çözerek dağıtık bloklar oluşturan bir blok madencisidir. Madenciler, mutabakat sorununu çözdükten sonra yeni bloklarını tüm ağa dağıtırlar (Miller et al. n.d.). Blok zincir teknolojisinin nesnelerin internetinde nasıl kullanılabileceğini anlamak için, temellerini ve merkeziyetsizliğini anlamak çok önemlidir.

Ağdaki madenciler, blok zincirin temel birimidir olan işlemi izlemektedir. Her işlem, işlemi gerçekleştirmek isteyen madenciyi tanımlamak için özel bir anahtarla başlar. İşlemin geçerliliğini doğrulamak için her bir özel anahtar, işlemin geçerliliğini doğrulamak için bir açık anahtarla bağlantılıdır. İşlem, ilk olarak Bitcoin'de iki finansal kuruluş arasındaki mali iletişimi kaydetmek için kullanılmıştır. Mülkiyet haklarını atamak ve programlanabilir olayları gerçekleştirmek de işlemler tarafından gerçekleştirilir (Segendorf 2014).

Her blok, dağıtık defterin bir parçası olan doğrulanmış bir işlem grubunu içerir. Kriptografik özetleme algoritması, blok başlığında her blok için özel bir özetleme değeri oluşturur (Wang et al. 2019). Her blok, bir önceki bloğa (örneğin Bitcoin blok zincirinde önceki bloğun özetleme değeri) bir bağlantı ve mutabakat sorusuna verilen cevaba sahiptir. Blok başlığı, gerekliliklere bağlı olarak zaman damgaları gibi diğer bileşenleri de içerebilir. Her ağ madencisinde, işlemlerin yerel bir kayıt olarak sıralı ve geriye bağlı bir blok listesi tutulur.

Blok zinciri, merkeziyetsiz bir teknoloji olarak işlemleri güvenli ve şeffaf bir biçimde kaydeden bir yapıya sahiptir ve bu süreç belirli adımlarla Şekil 3.3(Verma, Yadav, and Chandra 2022a)' deki gibi ilerler. İlk olarak, P2P bir işlem başlatılır ve bu işlem ağdaki diğer düğümlere iletilir. Bu işlem, kriptografik olarak imzalanarak işlem havuzuna eklenir. Ardından, ağdaki düğümler mutabakat algoritmalarını (örneğin, PoW veya PoS) kullanarak işlemi doğrular ve geçerliliğini teyit eder. Madenciler, işlem havuzundan bir işlem seçer, kullanıcıyı doğrular ve işlemi onaylar. Doğrulanan tüm işlemlerle birlikte yeni bir blok oluşturulur ve blok zincirine eklenir. Yeni oluşturulan blok, ağdaki düğümlere yayınlanır ve doğrulandıktan sonra blok zincire eklenir, böylece blok zincirde kalıcı hale gelir.

Blok zincirin tam bir kopyası, ağdaki tüm düğümler tarafından saklanır ve bu dağıtılmış defter yapısı, ağın güvenliğini sağlamak ve verilerin bozulmasına karşı koruma oluşturmak için kritik bir rol oynar. Sonuç olarak, bu süreç, merkezi bir otoriteye ihtiyaç duymadan güvenli, şeffaf ve değiştirilemez işlem kayıtları oluşturarak blok zincirin temel özelliklerini güçlendirir.



Şekil 3. 3: P2P ile blok zincirin gerçekleştirilmesi.

3.3. Blok zincir türleri

Blok zincirler genellikle diğer düğümlerin başlattığı işlemlere hangi düğümlerin erişebileceğini, bu işlemleri doğrulayabileceğini ve kimlik doğrulaması yapabileceğini belirleyen kurallara göre yapılandırılır (Guru et al. 2023),(Rahman et al. 2022). Blok zinciri yapılarının karşılaştırması Şekil 3.4' de verilmiştir. Bu bağlamda ortaya çıkan blok zincir türleri aşağıda kısaca açıklanmıştır.

3.3.1. Açık veya izinsiz blok zincirleri

Açık veya izinsiz blok zincirleri, herkesin katılımına ve ağ üzerinde işlem yapmasına izin veren merkeziyetsiz sistemlerdir (D. Li et al. 2020), (Guru et al. 2023). Bu tür blok zincirlerinde, herhangi bir kullanıcı ağa dahil olabilir, işlemleri doğrulayabilir ve yeni bloklar ekleyebilir (Ferdous et al. 2020). Şeffaflık ve güvenlik, mutabakat algoritmaları aracılığıyla sağlanır. Bitcoin ve Ethereum gibi popüler kripto para birimleri, bu model üzerinde çalışmaktadır (Santhoshi et al. n.d.), (Wust and Gervais 2018).

3.3.2. Özel veya izinli blok zincirleri

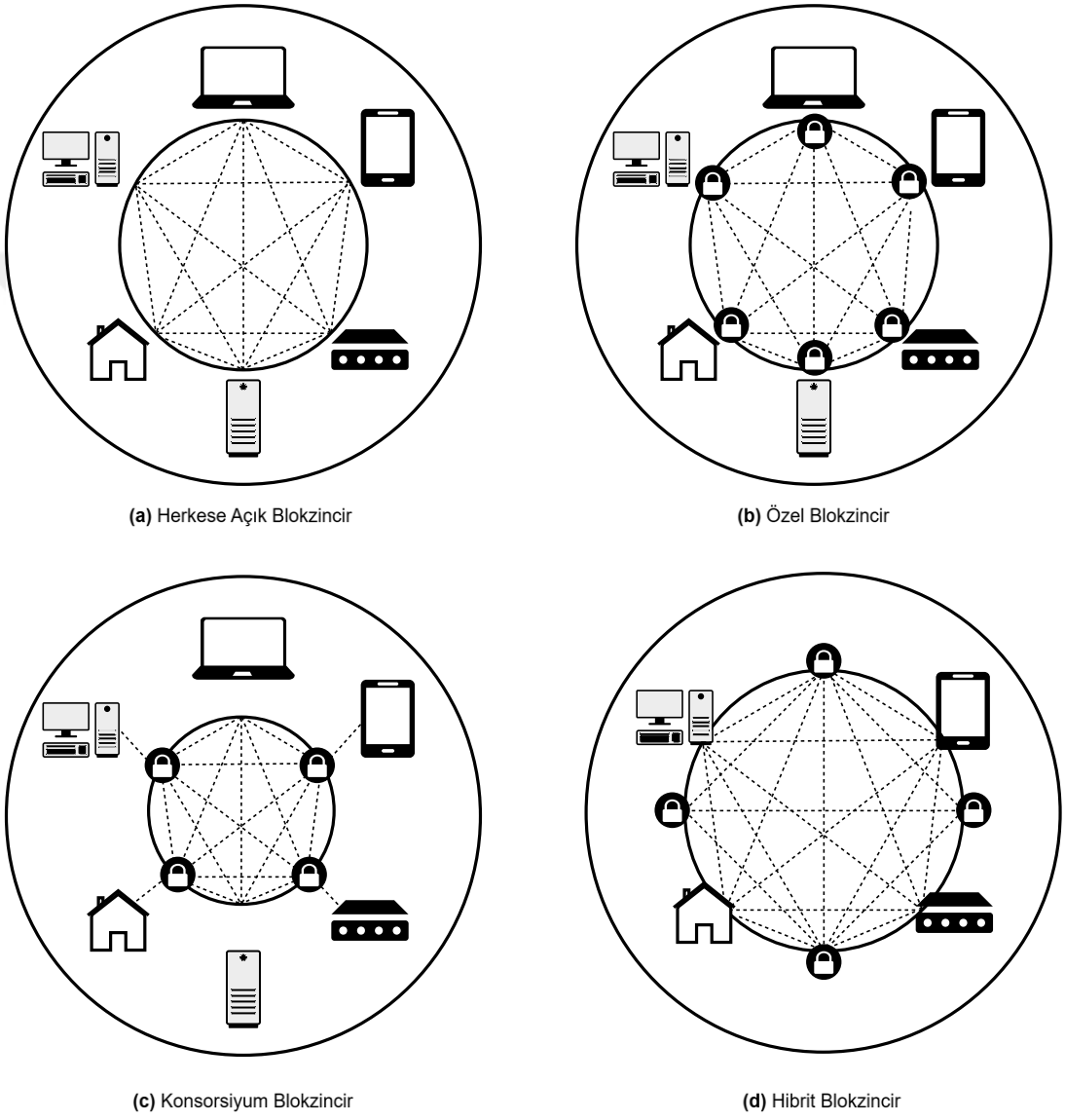
Özel veya izinli blok zincirleri, erişimin ve katılımın belirli bir organizasyon veya grup tarafından kontrol edildiği ağlardır (D. Li et al. 2020), (Guru et al. 2023), (Wust and Gervais 2018) . Bu sistemlerde, yalnızca yetkilendirilmiş kullanıcılar işlemleri görüntüleyebilir ve doğrulayabilir (Ferdous et al. 2020). Yüksek düzeyde gizlilik ve kontrol sağladığı için genellikle kurumsal çözümlerde tercih edilir. Örnek olarak, Hyperledger Fabric ve R3 Corda bu modele uygundur(Santhoshi et al. n.d.), (Wust and Gervais 2018).

3.3.3. Konsorsiyum blok zincirleri

Konsorsiyum blok zincirleri, birden fazla organizasyonun ortaklaşa yönettiği ve kontrol ettiği yarı özel ağlardır (Guru et al. 2023),(Verma et al. 2021). Katılım ve mutabakat mekanizmaları, önceden belirlenmiş bir grup tarafından yönetilir(Verma et al. 2021), (Santhoshi et al. n.d.). Bu yapı, farklı kurumların iş birliği yaparken hem veri paylaşımını hem de gizliliği dengeli bir şekilde yönetmelerine olanak tanır (Yao et al. 2023a). Bankacılık ve finans sektöründeki bazı ortak platformlar bu modeli kullanmaktadır.

3.3.4. Hibrit blok zincirleri

Hibrit blok zincirleri, açık ve özel blok zincirlerinin özelliklerini bir araya getiren karma sistemlerdir (Kim et al. 2022a). Bu tür ağlarda, bazı veriler ve işlemler herkese açıkken, diğerleri sadece yetkili kullanıcılar tarafından erişilebilir(Santhoshi et al. n.d.). Bu sayede hem şeffaflık hem de gizlilik ihtiyaçları aynı anda karşılanabilir. Hibrit modeller, çeşitli sektörlerin spesifik gereksinimlerine esnek çözümler sunar.



Şekil 3. 4*:* Blok zincir türleri.

Ayrıca, bir blok zinciri *görüntülemeye açık* veya *kullanıma özel* olabilir, ancak asla *kullanıma açık* değildir; aslında, izinli bir blok zinciri özeldir ve izinsiz bir blok zinciri kamuya açıktır. Bunun nedeni, *kamuya açık* teriminin bir blok zincirini kullanabilme

yeteneğine atıfta bulunmasıdır, işlem geçmişini görüntülemeye değil. Bu yanlış anlama bazı makalelerde görülebilir (Solat et al. 2021).

Blok zincirinin temel özelliklerinden biri, dağıtık defter teknolojisi, akıllı sözleşmeler, kriptografik algoritmalar ve mutabakatın birleşimi aracılığıyla bir şirket veya iş ağı içinde güven inşa etmektir. Defter, mevcut varlık durumunun yanı sıra tüm geçmiş işlem kayıtlarını da içerir. Ardışık işlemler her zaman birbirine güvenli bir şekilde bağlıdır, bu da etkili bozulmaları önler. Blok zincirinin değiştirilemezliği sayesinde, kayıt, birden fazla işletmeyi içeren iş operasyonlarıyla ilgili sorgular için tek güvenilir bilgi kaynağı olarak hizmet edebilir. Bu sayede blok zinciri, işletmeler arasındaki paylaşılan süreçleri otomatikleştirerek ve maliyetleri ile genel ağ karmaşıklığını azaltarak, bilginin keşfedilebilirliğini ve izlenebilirliğini artırır (D. Li et al. 2020).

Blok zinciri teknolojisinin en pratik uygulamaları, mutabakat, değişmezlik, köken ve kesinlik gibi özelliklerin güven oluşturduğu iş ağlarında görülür. Değişmezlik kavramı, işlem geçmişinin değiştirilemez olduğunu ifade eder. Her üyenin defterinin kopyasının diğer tüm kopyalarla eşleştiği ve blok zincirine kaydedilen işlemlerin gerçekten gerçekleştirilmiş olduğu garanti edildiği için, kesinlik iç huzuru sağlar. Köken, defterde tutulan herhangi bir varlığın bilinen bir başlangıcının olduğunu gösterir. Üyeler, üzerinde anlaşılmış bir onay politikasıyla belirtilen şekilde, defterdeki değişiklikleri yetkilendirmelidir. Bu hedef, sistemdeki katılımcıların işlemin geçerli olduğunu kabul ettiği ve defterin güncellenmiş durumunu onayladığı mutabakat yoluyla elde edilir (Zupan, Zhang, and Jacobsen 2017).

Blok zincir teknolojisi, erişim yapıları, merkeziyetsizlik seviyeleri, veri modelleri, ölçeklenebilirlik ve kullanım durumları dahil olmak üzere farklı türlerde çeşitli özellikler ve uygulamalar sunar. Araştırma, blok zincirin kripto para birimleri, finansal hizmetler, sağlık hizmetleri ve IoT (Monrat, Schelen, and Andersson 2019) gibi alanlardaki potansiyelini araştırmıştır. Blok zincir tipolojilerini değerlendirmek, performans sorunlarını ve güvenlik açıklarını ele almak için farklı analitik ve ampirik modeller kullanılmıştır (Rico-Peña, Arguedas-Sanz, and López-Martin 2023). IoT uygulamalarında, geleneksel merkezi yaklaşımların sınırlamalarını aşmak ve gelişmiş güvenlik ve gizlilik sunmak için blok zincir tabanlı erişim kontrol mekanizmaları önerilmiştir (Namane and Ben Dhaou 2022). Bu mekanizmalar, IoT sistemleri genelinde çeşitli uygulamalarla kısmen veya tamamen merkeziyetsiz olarak

sınıflandırılabilir (Namane and Ben Dhaou 2022), (Abdi et al. 2020). Ancak, ölçeklenebilirlik, dağıtım ve politika uygulama dahil olmak üzere IoT için merkezi olmayan erişim kontrolü geliştirmede zorluklar devam etmektedir (Abdi et al. 2020). Devam eden araştırmalar bu sorunları ele almaya ve blok zinciri teknolojisinin yeni uygulamalarını keşfetmeye devam etmektedir.

Tablo 3.1, blok zincir türlerinin erişim yapısı, merkeziyetsizlik seviyesi, veri modeli, ölçeklenebilirlik, kullanım senaryoları gibi özellikleri kapsar (Guidi and Michienzi 2023).

3.4. Doğrulama mekanizmaları

Bir blok zincir ağında, bir işlemin başarılı sayılabilmesi için iki kez doğrulanması gerekir. İki taraf arasındaki işlemi kolaylaştırmak için ilk aşamada bir akıllı sözleşme kullanılır. İşlemin mevcut durumu üzerinde anlaşmaya varmak için ise ikinci aşamada bir mutabakat algoritması kullanılır. Bu iki adım aşağıda detaylı bir şekilde ele alınacaktır.

Tablo 3. 2: Blok zincir türlerine göre karakteristik ve kullanım senaryoları karşılaştırması

Blok zincir Türü	**Ağ Erişimi**	**Merkeziyetsizlik**	**Veri Yapısı**	**Güvenlik**	**Kullanım Alanları**
Herkese açık	İzinsiz	Yüksek	Harcanmamış İşlem Çıkışı Modeli, Hesap Tabanlı Model	Düşük	Kripto Para, Akıllı Sözleşmeler
Konsorsiyum	İzinli	Orta	Hesap Tabanlı Model	Orta	Bankacılık, Tedarik Zinciri
Hibrit	İzinsiz/İzinli	Orta	Harcanmamış İşlem Çıkışı Modeli, Hesap Tabanlı Model	Yüksek	Nesnelerin interneti, Kurumsal Blok zincir Çözümleri
Özel	İzinli	Düşük	Hesap Tabanlı	Yüksek	Kurum içi çözümler

3.4.1. Akıllı sözleşme

Bilgisayar tabanlı işlem protokolleri olarak bilinen akıllı sözleşmeler, yasal olayları kaydetme, yönetme, yürütme ve sözleşme şartlarına uygun olarak faaliyetlerde bulunma yeteneğine sahiptir (Szabo 1997). Akıllı sözleşmelerin blok zincir teknolojisine dahil edilmesinin başlıca amaçları, dolandırıcılık kayıplarını azaltmak, güvenilir aracıların gerekliliğini ortadan kaldırmak ve yürütme ile beklenmeyen istisnalarla ilgili masrafları düşürmektir (Khan et al. 2021). Blok zincir teknolojisinin temeli akıllı sözleşmelerdir. İki taraf bir akıllı sözleşmeyi imzaladığında ve tüm şartlarını kabul ettiğinde, sözleşme anında blok zincire yüklenir ve belirli koşullar sağlandığında işlem tamamlanır (Imteaj, Hadi Amini, and Pardalos 2021).

3.4.2. Mutabakat

Anlaşma süreci, mutabakat olarak bilinir; her düğüm, mevcut bir blok zincire yeni bir blok ekleme yöntemini seçer. Blok zincir tabanlı herhangi bir sistemin temeli mutabakat mekanizmasıdır; bir sistem, onu yöneten mutabakat protokolü kadar güçlü ve güvenilirdir (Jones 2019). Yeni bir blok eklemek için anlaşmaya varmak amacıyla, katılımcı düğümler çeşitli mutabakat teknikleri kullanabilir (Cachin and Vukolić 2017a; Nguyen and Kim 2018). Mutabakat protokolleri, çalışma şekillerine göre geniş çapta gruplanabilir ve protokolleri, oylamaya dayalı veya kanıta dayalı olarak sınıflandırılabilir (Sankar, Sindhu, and Sethumadhavan 2017; Zhang and Lee 2020). Oylamaya dayalı protokoller izinli blok zincirlerde kullanılırken, kanıta dayalı mutabakat yöntemleri izinsiz blok zincirlerin temelini oluşturur. Bir mutabakat mekanizmasının hatalara toleransı da başka bir sınıflandırma şeması için temel teşkil eder. Blok zincir destekli sistemlerde en sık karşılaşılan üç hata, çökme (crash), Bizans ve çifte harcama hatalarıdır. Hata toleransı türüne göre, çökme ve Bizans hatalarına rağmen çalışabilen Mutabakat mekanizmalarıda mevcuttur. Bir sonraki bölümde, en bilinen yöntemler ve bunların karşılaştırmalı bir analizi ele alınacaktır. Dağıtık veya merkeziyetsiz ağlarda mutabakata ulaşmak için herhangi bir dizi kriter, örneğin çoğunluk yapısı, merkeziyetsiz yönetim, işlem gücü, Bizans hata toleransı vb. kullanılabilir. Bitcoin gibi kamuya açık blok zincirlerde, PoW fikri benimsenmiştir. Konsorsiyum ve özel blok zincirlerde, oylamaya dayalı mutabakat protokollerinin yerine, PoE, PoS, PoET, dPoS, PoI, PoStorage ve diğer hibrit kanıta dayalı mutabakat algoritmaları tercih edilmektedir (Verma, Yadav, and Chandra 2022a).

Mutabakat fonksiyonu, her blok zinciri düğümünün aynı mesaj üzerinde anlaşmasını sağlayan, en son bloğun zincire doğru bir şekilde eklendiğini güvence altına alan, her düğümün depoladığı mesajın aynı olduğunu ve *çatal saldırısına* yol açmayacağını garanti eden, ayrıca kötü niyetli saldırılara karşı savunma sağlayan bir mekanizmadır (Verma, Yadav, and Chandra 2022b),(Lin and Liao 2017) .

Mutabakat mekanizmaları, blok zincir teknolojisinin güvenliği, performansı ve güvenilirliği garanti eden önemli bileşenleridir (Zhou et al. 2023) (Yao et al. 2023b). Çeşitli mutabakat algoritmaları geliştirilmiştir; bunlardan PoW en yaygın olanıdır ancak enerji açısından verimsizdir (Aponte et al. 2021). Araştırmacılar, özellikle konsorsiyum blok zincirleri için performans ve verimlilik endişelerini gidermek amacıyla alternatif mekanizmalar önermişlerdir (Zhang et al. 2021). Bu mekanizmalar, güvenilirlik, performans ve güvenlik gibi kriterlere göre sınıflandırılabilir (Yao et al. 2023b). Bazı yaklaşımlar, performansı optimize etmek için güven değerlerine dayalı düğüm sınıflandırmasını içerir (Zhang et al. 2021). Giderek artan sayıda mutabakat tekniğini daha iyi anlamak için, farklı algoritmalar arasındaki davranış kalıplarını ve benzerlikleri ortaya çıkaran küme tabanlı sınıflandırma yöntemleri kullanılmıştır (Aponte et al. 2021). Blok zinciri teknolojisi gelişmeye ve çeşitli alanlarda uygulamalar bulmaya devam ettikçe, devam eden araştırmalar belirli kullanım durumlarına ve ağ türlerine göre uyarlanmış fikir birliği mekanizmaları geliştirmeye ve iyileştirmeye odaklanmaktadır(Zhou et al. 2023), (Yao et al. 2023b).

3.4.3. Mutabakat çalışma mantığı

Genel olarak, bir mutabakat algoritması, bir grup insanın farklı bakış açılarını paylaştığı ve bir sistem veya süreç için en iyi çözümü sunacak bir sonuç oluşturmak amacıyla karar verme tekniği olarak tanımlanır (Bhardwaj and Datta 2020a). Her grup üyesi, bir eylem planına ilişkin yapılan seçimleri güçlendirmek amacıyla düşüncelerini paylaşır. Basitçe söylemek gerekirse, bu, bir grupta ne yapılması gerektiğini belirlemenin bir yoludur. Gruptaki herkes bir öneride bulunmakta serbest olsada, çoğu kişi ihtiyaçlarına en uygun olan öneriyi destekleyecektir. Diğerleri bu kararı beğenip beğenmemelerine bakılmaksızın, bu kararla başa çıkmak zorundadır. Yalnızca tüm bileşenler uyum içinde çalışırsa, bir mutabakat sistemi başarılı olabilir. Bununla birlikte, bu sistemin sadece bir parçası arızalanırsa, sistemin bütünü çökebilir. Bir anlaşmaya varmak için bu blok zincir mutabakat modelleri kullanılır. Bununla birlikte,

ortak fikir birliği algoritmaları olmadan merkezi olmayan bir sistem olamaz. Düğümlerin birbirlerine güvenip güvenmemelerinin bile önemi yoktur. Belirli ilkelere dayanarak hareket etmeleri ve birlikte bir karara varmaları gerekecektir. Bunu yapmak için tüm mutabakat algoritmalarını kontrol etmek gerekir. Fikir birliği algoritmaları, blok zincir ağlarının çok yönlü olmasının nedenidir. Bununla birlikte, blok zincir fikir birliği algoritmasının hem avantajları hem de dezavantajları her zaman algoritmanın mükemmelliğini değiştirebilir (Bhardwaj and Datta 2020a).

Blok zincir tabanlı sistemlerin güvenliğini sağlamak için akıllı sözleşmelerin doğru çalışması çok önemlidir. Kriptografik kurallar ve matematik, blok zincirin sisteme güven vermesini sağlar. Bu güven, temel protokollerin düzenli olarak yürütülmesine ve uygulanmasına dayanır. Sisteme güven vermek için mutabakat protokolleri kullanılır (Verma et al. 2022a), (De Filippi, Mannan, and Reijers 2020b).

3.4.4. Mutabakat türleri

Mutabakat algoritmaları, IoT ve akıllı şebekeler dahil olmak üzere çeşitli uygulamalarda blok zinciri güvenliği ve performansı için çok önemlidir (Ferrag and Shu 2021), (Zhang, Zhu, and Ali 2024). Bu algoritmalar ölçeklenebilirlik, enerji verimliliği ve işlem hızı açısından zorluklarla karşı karşıyadır (Alam 2023a). Araştırmacılar, gecikme sürelerini, verimlerini, hesaplama maliyetlerini ve ölçeklenebilirliklerini karşılaştırarak farklı mutabakat mekanizmalarını analiz ettiler (Ferrag and Shu 2021). Blok zinciri sistemlerinin performans değerlendirmesi, mutabakat süresi, saldırı direnci ve işlem işleme hızı gibi ölçütlerin incelenmesini içerir (Touloupou et al. 2022a), (Zhang et al. 2024). Oyun teorisi, BAN mantığı ve simülasyon çerçeveleri dahil olmak üzere çeşitli araçlar ve teknikler, blok zinciri güvenliğini ve performansını değerlendirmek için kullanılır (Ferrag and Shu 2021), (Touloupou et al. 2022a). Mevcut fikir birliği algoritmalarındaki sınırlamaları ele almak için, akıllı şebeke uygulamalarında işbirlikçi PoW için kümeleme düğümleri gibi yeni yaklaşımlar geliştirilmektedir (Zhang et al. 2024). Bu çabalar, fikir birliği süresi, enerji verimliliği ve ölçeklenebilirlik açısından blok zinciri performansını iyileştirmeyi amaçlamaktadır.

3.4.5. Erişim modellerine göre mutabakat türleri

Blok zincir ağları, erişim ve yönetim modellerine dayalı olarak genel anlamda iki ana kategoriye ayrılmaktadır: izinsiz ve izinli blok zincirler. İzinsiz blok zincirler, merkeziyetsizliği ve açıklığı esas alan sistemlerdir ve genellikle kanıta dayalı mutabakat mekanizmaları kullanırlar. Buna karşılık, izinli blok zincirler daha kapalı, kontrollü ve katılımcıların önceden belirlendiği yapılar olup, oylamaya dayalı mutabakat protokollerini tercih ederler. İzinli yapılar kendi içinde iki alt gruba ayrılmaktadır: federatif (konsorsiyum tabanlı) sistemler ve özel/hibrit sistemler. Bu sınıflandırma, farklı blok zincir türlerinin güvenlik, ölçeklenebilirlik ve performans ihtiyaçlarına göre nasıl şekillendiğini ve optimize edildiğini ortaya koymaktadır (Chenchev 2023a).

İzinsiz blok zincirler, herkesin katılımına açık ve merkeziyetsiz bir yapıya sahiptir. Bu tür blok zincirlerde işlemleri doğrulamak için kanıta dayalı mutabakat protokolleri kullanılmaktadır. Bu protokoller arasında şunlar yer almaktadır; PoS, katılımcıların sahip oldukları kripto para miktarına dayalı olarak işlemleri doğruladığı bir modeldir (Verma et al. 2022a) (King and Nadal 2012) (Islam et al. 2023) (Chenchev 2023b). DPoS, PoS'a benzer, ancak işlemler belirli temsilciler tarafından doğrulanır (Verma et al. 2022a) (Islam et al. 2023). Ouroboros, güvenliği artırmayı hedefleyen PoS tabanlı bir mutabakat algoritmasıdır (Aragon et al. 2022). PoD, katılımcıların belirli miktarda varlık yatırarak işlemleri doğruladığı bir modeldir (Hasan and Salah 2018). PoB, katılımcılar, kripto paralarını yakarak işlemleri doğrular (Verma et al. 2022a) (Chenchev 2023b). PoCA, bir coin'in ağda ne kadar süre tutulduğuna dayalı bir doğrulama mekanizmasıdır (Shi et al. 2023) (Chenchev 2023b). Casper, Ethereum'un PoS'a geçiş sürecinde kullanılan protokoldür (Al Ahmad, Al-Saleh, and Al Masoud 2018) (Chenchev 2023b). Casper CBC ve Casper FFG, Ethereum 2.0 için geliştirilen, güvenliği artıran iki sonlandırma mekanizmasıdır (Buterin et al. 2019) (Buterin et al. 2020). PoW, hesaplama gücüne dayalı, işlemleri doğrulamak için kullanılan geleneksel bir modeldir (S. Verma et al., 2022a) (Nakamoto n.d.) (Islam et al. 2023) . PoX, kripto varlıkların başka bir blok zincire transferine dayalı bir protokoldür (Pbc 2020). PoC, katılımcıların depolama alanlarına dayalı olarak işlemleri doğruladığı bir modeldir (Verma et al. 2022a). PoET, katılımcıların rastgele belirlenmiş bir süre beklemesini gerektiren bir modeldir (Verma et al. 2022a) (Islam et al. 2023). PoA, yetkilendirilmiş düğümlerin işlemleri doğruladığı bir modeldir (Anon 2023) (Islam et

al. 2023). PoAc, Hem PoW hem de PoS unsurlarını birleştiren bir modeldir (Islam et al. 2023) (Ometov et al. 2020). PoSp, depolama alanı kullanımına dayalı bir mutabakat mekanizmasıdır (Dziembowski et al. n.d.). DPoW, belirli bir iş yüküne dayalı PoW varyantıdır (Bazzanella and Gangemi 2023). PoI, kullanıcıların ağdaki aktivitesi ve itibarı temel alınarak işlemleri doğruladığı bir mekanizmadır (Verma et al. 2022a). PoRx, kullanıcıların itibarına dayalı bir doğrulama mekanizmasıdır (Islam et al. 2023).

İzinli blok zincirler, izinli blok zincirler, belirli yetkilendirilmiş kullanıcıların katılımına açık olan daha kapalı ve kontrollü sistemlerdir. Bu tür sistemlerde genellikle oylamaya dayalı mutabakat protokolleri kullanılır.

Konsorsiyum tabanlı blok zincirler, birden fazla kuruluş tarafından yönetilen blok zincirlerdir. RPCA, Ripple ağı tarafından kullanılan mutabakat algoritmasıdır (Schwartz, Youngs, and Britto n.d.) (Islam et al. 2023). SCP, Stellar ağında işlemleri hızlı ve güvenli bir şekilde doğrulayan protokoldür (Mazi`eres and Mazi`eres n.d.).

Özel/Hibrit blok zincirler, tek bir kuruluş veya birkaç organizasyon tarafından yönetilen sistemlerdir. PoET, izinsiz PoET'ye benzer, ancak izinli ağlarda kullanılır (Verma et al. 2022a) (Islam et al. 2023). PoL, katılımcıların şans faktörüne dayalı olarak işlemleri doğrulamalarını sağlar (Ogawa, Kima, and Miyaho 2018). PBFT, Bizans hatalarına karşı dirençli bir mutabakat mekanizmasıdır (Verma et al. 2022a) (Islam et al. 2023). Raft, daha basit bir oylama sistemi olup, genellikle küçük kapalı ağlarda kullanılır (Verma et al. 2022a) (Anon 2003). PoA, yetkilendirilmiş düğümlerin işlemleri doğruladığı modeldir (Islam et al. 2023). PoAc, Hem PoW hem de PoS unsurlarını birleştirir (Islam et al. 2023).

3.4.6. Kullanım türüne göre mutabakat türleri

Blok zincirlerde kullanılan mutabakat algoritmaları genel olarak iki ana kategori altında sınıflandırılmaktadır: kanıta dayalı ve oylamaya dayalı protokoller. Her bir kategori, farklı güvenlik seviyeleri, performans beklentileri ve enerji verimliliği açısından çeşitli algoritmalar içermektedir. Kanıta dayalı protokoller daha çok kamuya açık ve merkeziyetsiz blok zincirlerde yaygın olarak tercih edilirken, oylamaya dayalı protokoller ise genellikle izinli, kapalı sistemlerde kullanılmaktadır. Bu ayrım, blok zincir ağlarının sahip olduğu farklı güvenlik, ölçeklenebilirlik ve işlem hacmi gereksinimlerine uygun olarak mutabakat sağlama yaklaşımlarının nasıl optimize edildiğini ortaya koymaktadır (Chenchev 2023a)**.**

Kanıta dayalı mutabakat algoritmaları, işlemleri doğrulamak ve blok eklemek için belirli bir kaynağın (hesaplama gücü, teminat(stake) edilen varlık, depolama kapasitesi vb.) kullanıldığı algoritmalardır. Bu kategori, blok zincir ağlarında en yaygın kullanılan mutabakat mekanizmalarından bazılarını içerir:

PoW. Credit-Based, PoW'un kredi bazlı bir varyantıdır (Chenchev 2023b). Casper FFG, Ethereum'un Proof of Stake geçişinde kullanılan ve PoW ile PoS'yi birleştiren bir protokoldür. dPoW, PoW'un zaman gecikmesiyle çalışan bir versiyonudur (Chenchev 2023b). PoMW, yapılan işin anlamlı olmasını sağlayan bir protokoldür (ASANUMA and ISOBE 2023). PoWT, belirli bir süre boyunca yapılan işin doğrulama için kullanıldığı bir PoW varyantıdır (Chenchev 2023b) ve PoX, PoS, DPoS, Ouroboros, PoD, PoB, PoCA, Casper CBC, Casper FFG, PoC ve PoET gibi diğer konsensüs algoritmalarıyla birlikte değerlendirilmektedir.

Oylamaya dayalı mutabakat algoritmaları, işlemleri doğrulamak için düğümler arasında bir oylama süreci yürütülen algoritmalardır. Bu kategori genellikle kapalı veya izinli blok zincirlerde tercih edilir: BFT, Bizans hata toleransına sahip, ağdaki kötü niyetli düğümlere karşı dayanıklı bir mutabakat algoritmasıdır. BFT-SMaRt, HotStuff, Cerberus, BFT'nin farklı versiyonlarıdır, her biri farklı güvenlik ve performans kriterlerine odaklanır (Cäsar et al. n.d.). DBFT, temsilciler tarafından kullanılan bir Bizans hata toleransına dayalı algoritmadır. PBFT, ağdaki kötü niyetli düğümlere karşı dayanıklı, pratik bir Bizans hata toleransı protokolüdür. RBFT, PBFT'nin daha fazla hata toleransı sunan bir versiyonudur (Aublin, Mokhtar, and Quema 2013). Query/Update ve Hybrid Quorum , PBFT'nin hibrit ve sorgu tabanlı varyantlarıdır (Abd-El-Malek et al. 2005) (Clement et al. n.d.). Zyzzyva, Aardvark, Tendermint, Bizans hata toleransına dayalı diğer mutabakat protokolleridir (Buchman 2016; Clement et al. n.d.; Kotla et al. 2009). Raft, kapalı ve izinli blok zincirlerde basit bir oylama süreci aracılığıyla kullanılan bir mutabakat algoritmasıdır (Anon 2003). Paxos, oylamaya dayalı mutabakat algoritmalarından biri olup, çeşitli varyantları bulunmaktadır (Lamport 2001): Egalitarian Paxos, Multi Paxos, Fast Paxos, Paxos'un performans ve güvenlik açısından optimize edilmiş versiyonlarıdır (Lamport 2001). Random Exponential Backoff Paxos ve Mencius, Paxos'un hata toleransını artırmayı amaçlayan versiyonlarıdır (Lamport 2001).

3.4.7. Mutabakat algoritmalarının performans ve yapısal karakteristik karşılaştırması

Mutabakat algoritmaları, blok zinciri güvenliği ve merkeziyetsizliği için çok önemlidir. Dağıtık sistemlerde güvenmeyen düğümler arasında anlaşmayı sağlarlar (Bhardwaj and Datta 2020b). Ancak bu algoritmalar, ağın karma gücünün yarısından fazlasını kontrol eden saldırganların işlemleri manipüle edebileceği ve paraları iki kez harcayabileceği %51 saldırısı gibi zorluklarla karşı karşıyadır (Sayeed and Marco-Gisbert 2019). Her biri verim, ölçeklenebilirlik, gecikme ve enerji verimliliği açısından farklı takaslara sahip çeşitli mutabakat mekanizmaları mevcuttur (Alam 2023b). Arızalı veya kötü amaçlı bileşenlerle bile mutabakat sistemlerinin doğru şekilde çalışması gerektiğinden Bizans hata toleransı önemli bir husustur (Bhardwaj and Datta 2020b). Ölçeklenebilirlik sorunlarını ele almak ve merkeziyetsizliği sürdürmek için bazı yaklaşımlar mutabakatı seçili düğümlere devreder. Merkezi olmayan aracılar arasında BFT, blok başına rastgele fikir birliği düğümleri seçmek için bir yöntem öneriyor ve blok zincirinin güvenlik, ölçeklenebilirlik ve merkeziyetsizlik üçlemesini çözmeyi amaçlıyor (Oh et al. 2020).

Tablo 3.2, algoritmaların performans, enerji verimliliği, gecikme (latency), ölçeklenebilirlik, TPS ve kullanılan özetleme algoritmaları gibi özellikleri içerir.

Tablo 3. 3: Mutabakat algoritmalarının performans ve yapısal karakteristik karşılaştırması

Algoritma	Blok zincir Türü	Enerji Verimliliği	Gecikme	Ölçeklenebilirlik	Saniye Başına İşlem	Özetlem Algoritması	Kullanım Alanları
PoW	Genel	Düşük	Yüksek	Düşük	7-25	SHA-256	Kripto Paralar
PoS	Genel	Orta	Orta	Orta	1000+	SHA-3	Akıllı Sözleşmeler
DPoS	Genel	Yüksek	Düşük	Yüksek	100,000+	SHA-256, RIPEMD	Hızlı İşlem Ağları
PoA	İzinli	Yüksek	Düşük	Orta	1000	SHA-3	Kurumsal Blok zincir
PBFT	İzinli	Yüksek	Düşük	Yüksek	1000+	SHA-256	Finansal Blok zincir
PoET	İzinli	Yüksek	Düşük	Orta	1000+	SHA-256	IoT, Kurumsal Blok zincir
Raft	İzinli	Yüksek	Düşük	Orta	1000+	CRC32	Merkezi olmayan veri yönetimi
Paxos	İzinli	Orta	Orta	Düşük	1000+	MD5	Dağıtık sistemlerde veri tutarlılığı
Tendermint	Genel/Özel	Yüksek	Düşük	Yüksek	10,000+	SHA-256	Blok zincir Ağları
IBFT	İzinli	Orta	Orta	Orta	1000+	SHA-3	Hyperledger, Corda

Kaynak: (Anon 2018), (Bosamia and Patel 2020), (Merrad et al. 2022), (Merrad et al. 2022), (Merrad et al. 2022)

3.4.8. Güvenlik, dayanıklılık ve yönetim özellikleri karşılaştırması

Mutabakat algoritmaları, blok zinciri teknolojisi için verimliliği, güvenliği ve enerji tüketimini etkileyerek kritik öneme sahiptir. İş Kanıtı ve Hisse Kanıtı yaygın algoritmalardır, ancak hibrit yaklaşımlar da ortaya çıkmaktadır (Hussein, Salama, and El-Rahman 2023). CRSM, düğümleri eşzamanlı mutabakat alanlarına bölerek verimliliği artırır (Hu et al. 2020). Benzer şekilde, HDPoR algoritması, yenilenebilir enerji işlemlerini yönetmek için paralel bilgi işlem yeteneklerini geliştirir (Huh and Kim 2019). Enerji verimliliği, özellikle dağıtılmış enerji ticareti gibi gerçek zamanlı uygulamalar için önemli bir endişe kaynağıdır (Hu et al. 2020).

Tablo 3.3, algoritmaların güvenlik, %51 saldırısına dayanıklılık, Bizans hata dayanıklılığı, çökme toleransı, merkeziyetsizlik ve düğüm kimliği gibi güvenlik ve yönetim odaklı özelliklerini içerir.

Blok zinciri protokol performansını değerlendirmek ve ölçeklenebilirlik ve güç tüketimi gibi zorlukları ele almak için kıyaslama çerçeveleri ve simülatörler geliştirilmektedir (Touloupou et al. 2022b). Blok zinciri teknolojisi gelişmeye devam ettikçe, merkeziyetsizlik, güvenlik ve enerji verimliliği gibi faktörleri dengeleyen mutabakat algoritmasının seçimi kritik önemini korumaktadır (Hussein et al. 2023).

Tablo 3. 4: Güvenlik, dayanıklılık ve yönetim özellikleri karşılaştırması

Algoritma	%51 Saldırısına Dayanıklılığı	Bizans Hata Dayanıklılığı	Çökme Dayanıklılığı	Güvenlik Seviyesi	Merkeziyetsizlik	Düğüm Kimliği	Yönetim Yapısı
PoW	Düşük	Yok	Yok	Yüksek	Yüksek	Pseudonymous	Tamamen açık
PoS	Orta	Yok	Yok	Orta	Orta	Gerçek Kimlik (Stake sahipleri)	Stake sahipleri
DPoS	Orta	Orta	Yok	Yüksek	Orta	Seçilmiş Temsilciler	Delegeler yönetir
PoA	Yüksek	Orta	Yok	Yüksek	Düşük	Gerçek Kimlik (Seçilmiş doğrulayıcılar)	Seçilmiş doğrulayıcılar
PBFT	Yüksek	Yüksek	Yok	Yüksek	Düşük	İzinli Düğümler	Seçilmiş doğrulayıcılar
PoET	Yüksek	Yüksek	Yok	Yüksek	Düşük	İzinli Düğümler	Özel kullanıcılar
Raft	Yok	Yok	Var	Orta	Düşük	İzinli Düğümler	Merkezi yapı
Paxos	Yok	Yok	Var	Yüksek	Düşük	İzinli Düğümler	Merkezi yapı
Tendermint	Yüksek	Yüksek	Yok	Yüksek	Orta	İzinli veya İzinsiz	Delegeler yönetir
IBFT	Yüksek	Yüksek	Yok	Yüksek	Orta	İzinli Düğümler	Delegeler yönetir

Kaynak: (Anon 2018), (Bender and Benedict n.d.), (Bosamia and Patel 2020), (Merrad et al. 2022), (Merrad et al. 2022).

3.4.9. Enerji verimliliği, dil ve yürütme özellikleri karşılaştırması

Tablo 3.4' de algoritmaların enerji verimliliği, çevresel etki, programlama dili ve yürütme modeli gibi özellikler karşılaştırılmıştır.

Tablo 3. 5: Enerji verimliliği, dil ve yürütme özellikleri karşılaştırması

Algoritma	Enerji Verimliliği	Çevresel Etki	Programlama Dili	Yürütme Modeli
PoW	Düşük	Yüksek	C++, Python	Madencilik, EVM
PoS	Orta	Orta	Solidity, Vyper	Doğrulayıcı Seçimi
DPoS	Yüksek	Düşük	C++, Python	Delege Edilmiş Oylama
PoA	Yüksek	Düşük	Solidity	Otorite Seçim
PBFT	Yüksek	Orta	Go, Java	Bizans Anlaşması
PoET	Yüksek	Düşük	Python, C++	Güvenilir Yürütme
Raft	Yüksek	Düşük	Go, Python	Lider Seçimi
Paxos	Orta	Orta	Java, Python	Mutabakat Oylaması
Tendermint	Yüksek	Düşük	Go, Python	Bizans Anlaşması
IBFT	Orta	Orta	Solidity	Bizans Anlaşması

Kaynak: (Anon 2018), (Bosamia and Patel 2020), (Merrad et al. 2022),

Tablo 3.5, Tablo 3.6, Tablo 3.7, Tablo 3.8, Tablo 3.9, Tablo 3.10, Tablo 3.11, ve Tablo 3.12' da yer alan veriler, endüstriyel otomasyon sistemlerinde kullanılan dört farklı blok zinciri türüne (açık kaynaklı, kapalı kaynaklı, hibrit ve konsorsiyum) göre en sık kullanılan platformları ve bunların çeşitli özelliklerini karşılaştırmaktadır. (Anon n.d.-b; Bellaj et al. 2022; Cachin and Vukolić 2017b; Casino, Dasaklis, and Patsakis 2019; Dib et al. 2018; Ferdous et al. n.d.; Rathore, Mohamed, and Guizani 2020; Salimitari and Chatterjee 2018; Thakur and Kulkarni 2017; Zou et al. 2020).

Her platformun mutabakat algoritması, kullanım alanları, saniye başına, güvenlik, ölçeklenebilirlik, enerji verimliliği, akıllı sözleşmelerin desteklenmesi, merkeziyetsiz uygulamaların geliştirilip geliştirilebilmesi, açık kaynak kodlu olup olmaması ve açık API'ler sunulup sunulmadığı gibi önemli özellikleri içermektedir. Ayrıca açık kaynaklı blok zincir platformları arasında Ethereum ve Hyperledger Sawtooth gibi platformlar, akıllı sözleşmelerin ve merkeziyetsiz uygulamaların geliştirilmesine olanak tanırken, IOTA daha çok IoT odaklı ve enerji verimliliği yüksek bir platformdur. Kapalı

kaynaklı blok zincirler arasında Corda ve Quorum gibi platformlar, yüksek güvenlik ve enerji verimliliği sunarken, genellikle kurumsal ve finansal uygulamalar için tercih edilmektedir. Hibrit blok zincirler hem merkeziyetsizliği hem de güvenliği dengelemeye çalışan platformlar olup, Dragonchain ve XinFin gibi platformlar akıllı sözleşmeler ve kurumsal uygulamalarda yaygın olarak kullanılmaktadır. Konsorsiyum blok zincirleri ise birden fazla kuruluşun iş birliği yaptığı sistemler olup, Hyperledger Fabric ve VeChain gibi platformlar tedarik zinciri ve lojistik gibi sektörlerde güçlü bir izlenebilirlik ve güvenlik sağlamaktadır.

Tablo 3. 6: Açık kaynaklı blok zincirlerinin performans ve güvenlik özellikleri.

Blok Zincir Türü	**Platform**	**Mutabakat Algoritması**	**Kullanım Alanları**	**Saniye Başına İşlem**	**Güvenlik**	**Ölçeklenebilirlik**
Açık Kaynaklı Blok Zincirler	**Ethereum**	PoS	Akıllı sözleşmeler, finans, tedarik zinciri	~15 TPS	Orta	Orta
	Hyperledger Sawtooth	PoET	IoT, tedarik zinciri, endüstriyel otomasyon	~1000 TPS	Yüksek	Yüksek
	IOTA	DAG	IoT, makine ödemeleri	Yüksek (Teorik olarak sınırsız)	Orta	Çok Yüksek

Kaynak: (Merrad et al. 2022), (Makhdoom, Abolhasan, and Ni 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas, Caprolu, and Di Pietro 2024),(Abbas, Caprolu, and Di Pietro 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 7 Açık kaynaklı blok zincirlerinin işlevsel özellikleri.

Blok Zincir Türü	Platform	Enerji Verimliliği	Akıllı Sözleşmeler	Dağıtık Uygulamalar	Açık Kaynak	Açık APIs
Açık Kaynaklı Blok Zincirler	**Ethereum**	Yüksek	Evet	Evet	Evet	Evet
	Hyperledger Sawtooth	Orta	Evet	Hayır	Evet	Evet
	IOTA	Yüksek	Hayır	Hayır	Evet	Evet

Kaynak: (Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 8: Kapalı kaynaklı blok zincirlerinin performans ve güvenlik özellikleri.

Blok Zincir Türü	Platform	Mutabakat Algoritması	Kullanım Alanları	Saniye Başına İşlem	Güvenlik	Ölçeklenebilirlik
Kapalı Kaynaklı Blok Zincirler	**Corda**	Noter Tabanlı Mutabakat	Finans, kurumsal otomasyon	~170 TPS	Çok Yüksek	Orta
	Quorum	PoA, IBFT	Bankacılık, finansal hizmetler, kurumsal çözümler	~100 TPS	Yüksek	Orta
	Ripple	RPCA	Finansal işlemler, bankalar arası para transferi	~1500 TPS	Yüksek	Orta

(Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 9: Kapalı kaynaklı blok zincirlerinin işlevsel özellikleri.

Blok Zincir Türü	Platform	Enerji Verimliliği	Akıllı Sözleşmeler	Dağıtık Uygulamalar	Açık Kaynak	Açık APIs
Kapalı Kaynaklı Blok Zincirler	**Corda**	Yüksek	Hayır	Hayır	Hayır	Hayır
	Quorum	Yüksek	Evet	Evet	Hayır	Evet
	Ripple	Yüksek	Hayır	Hayır	Hayır	Evet

(Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 10: Hibrit blok zincirlerinin performans ve güvenlik özellikleri.

Blok Zincir Türü	Platform	Mutabakat Algoritması	Kullanım Alanları	Saniye Başına İşlem	Güvenlik	Ölçeklenebilirlik
Hibrit Blok Zincirler	**Polkadot**	NPoS	Akıllı sözleşmeler, IoT, tedarik zinciri	~1000 TPS	Yüksek	Yüksek
	XinFin (XDC Network)	DPoS	Ticaret ve finans, IoT	~2000 TPS	Yüksek	Yüksek
	Kadena	PoW, Chainweb	Akıllı sözleşmeler, endüstriyel otomasyon	~480,000 TPS (Teorik)	Yüksek	Yüksek

Kaynak: (Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 11: Hibrit blok zincirlerinin işlevsel özellikleri.

Blok Zincir Türü	Platform	Enerji Verimliliği	Akıllı Sözleşmeler	Dağıtık Uygulamalar	Açık Kaynak	Açık APIs
Hibrit Blok Zincirler	**Polkadot**	Yüksek	Evet	Evet	Evet	Evet
	XinFin (XDC Network)	Yüksek	Evet	Evet	Evet	Evet
	Kadena	Orta	Evet	Hayır	Evet	Evet

Kaynak: (Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 12: Konsorsiyum blok zincirlerinin performans ve güvenlik özellikleri.

Blok Zincir Türü	Platform	Mutabakat Algoritması	Kullanım Alanları	Saniye Başına İşlem	Güvenlik	Ölçeklenebilirlik
Konsorsiyum Blok Zincirler	**R3 Corda**	Noter Tabanlı Mutabakat	Finans, bankacılık, otomasyon sistemleri	~170 TPS	Çok Yüksek	Orta
	Hyperledger Fabric	PBFT	Tedarik zinciri, kurumsal çözümler	~3500 TPS	Yüksek	Yüksek
	VeChain	PoA	Tedarik zinciri, lojistik, endüstriyel uygulamalar	~10,000 TPS	Yüksek	Yüksek

Kaynak: (Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3. 13 Konsorsiyum blok zincirlerinin işlevsel özellikleri.

Blok Zincir Türü	Platform	Enerji Verimliliği	Akıllı Sözleşmeler	Dağıtık Uygulamalar	Açık Kaynak	Açık APIs
Konsorsiyum Blok Zincirler	R3 Corda	Yüksek	Hayır	Hayır	Hayır	Hayır
	Hyperledger Fabric	Orta	Evet	Hayır	Evet	Evet
	VeChain	Yüksek	Evet	Hayır	Hayır	Evet

Kaynak: (Merrad et al. 2022), (Makhdoom et al. 2018), (Parssinen et al. 2018), (Hanis, Rasid, and Blok zincir n.d.), (Clincy and Shahriar 2019), (Burdges et al. 2020; Clincy and Shahriar 2019), (Abbas et al. 2024),(Abbas et al. 2022), (Wang et al. 2021), (XinFin (XDC) Hybrid Blok zincir R&D Team 2021), (Cachin and Vukolić 2017c), (Baresi et al. 2022), (Chitra et al. 2019)

Tablo 3.13 ve Tablo 3.14, blok zincir mutabakat algoritmalarının farklı blok zincir türlerine (açık, kapalı, hibrit ve konsorsiyum) göre nasıl performans gösterdiğini kapsamlı bir şekilde karşılaştırmaktadır. Bu analizde, enerji tüketimi, teşvik yapısı, performans, blok oluşturma süresi, ölçeklenebilirlik, güvenlik riskleri ve kullanım alanları gibi kriterler göz önüne alınmıştır. Her blok zincir türü, belirli bir amaca hizmet etmekte ve kullanılan mutabakat algoritmaları da bu amaçlara uygun olarak seçilmektedir. Açık blok zincirler, merkeziyetsizlik ve güvenliği önceliklendirir. Bu tür blok zincirlerde PoW ve PoS gibi enerji yoğun, ancak güvenilir algoritmalar yaygın olarak kullanılır. Ancak bu algoritmalar, düşük ölçeklenebilirlik ve yüksek enerji tüketimi gibi sınırlamalar taşır (Sharma and Lal 2020), (Anon 2019a), (Anon 2019b).

Tablo 3. 14: Açık ve kapalı blok zincirlerinde mutabakat algoritmalarının performansı.

Blok zincir Türü	Algoritma	Algoritma Türü	Enerji Tüketimi	Teşvik Yapısı	Performans	Blok Oluşturma Süresi	Ölçeklenebilirlik	Çifte Harcama Riski	Byzantine Hata Toleransı	Kullanım Alanları
Açık Blok Zincir	PoW	Proof-based	Yüksek	Madencilik ödülü, işlem ücreti	<100 TPS	>10s	Düşük	Var	Yok	Kripto para, Akıllı sözleşmeler
	PoS	Proof-based	Orta	Madencilik ödülü	<1000 TPS	<10s	Güçlü	Yok	Var	Kripto para, Akıllı sözleşmeler
	DPoS	Proof-based	Orta	Madencilik ödülü	<1000 TPS	<10s	Güçlü	Yok	Var	Kripto para, BitShares
Kapalı Blok Zincir	PBFT	Vote-based	Düşük	Yok	>1000 TPS	<1s	Orta	Yok	Var	Kurumsal uygulamalar, Akıllı sözleşmeler
	Raft	Vote-based	Düşük	Yok	>10k TPS	>1s	Güçlü	Yok	Var	Kurumsal ve genel uygulamalar
	PoET	Proof-based	Çok düşük	Yok	>1000 TPS	<1s	Güçlü	Yok	Var	IoT, Kurumsal uygulamalar

Kaynak: (Anon 2018), (Merrad et al. 2022), (Yadav and Singh 2020)

Tablo 3. 15: Hibrit ve Konsorsiyum blok zincirlerinde mutabakat algoritmalarının performansı.

Blok zincir Türü	Algoritma	Algoritma Türü	Enerji Tüketimi	Teşvik Yapısı	Performans	Blok Oluşturma Süresi	Ölçeklenebilirlik	Çifte Harcama Riski	Byzantine Hata Toleransı	Kullanım Alanları
Hibrit Blok Zincir	PoS	Proof-based	Orta	Madencilik ödülü	<1000 TPS	<10s	Güçlü	Yok	Var	Akıllı sözleşmeler, IoT
	DPoS	Proof-based	Orta	Madencilik ödülü	<1000 TPS	<10s	Güçlü	Yok	Var	Hibrit sistemler, IoT
	PoC	Proof-based	Düşük	Madencilik ödülü	<1000 TPS	>10s	Orta	Yok	Var	Veri depolama, Akıllı sözleşmeler
Konsorsiyum Blok Zincir	PBFT	Vote-based	Düşük	Yok	>1000 TPS	<1s	Orta	Yok	Var	Konsorsiyum yapılar, Kurumsal
	Paxos	Vote-based	Düşük	Yok	>10k TPS	>1s	Güçlü	Yok	Var	Dağıtık sistemler, Kurumsal
	Tendermint	Vote-based	Düşük	Yok	>1000 TPS	<1s	Güçlü	Yok	Var	Dağıtık ağlar, Kurumsal

Kaynak: (Anon 2018), (Merrad et al. 2022), (Yadav and Singh 2020)

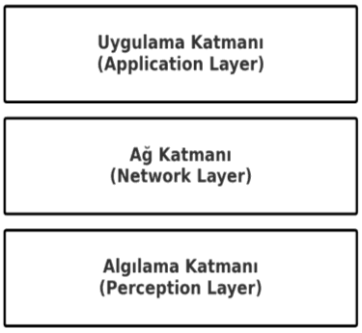
Kapalı blok zincirler, sınırlı katılımcı erişimine sahip, daha kontrollü ağlardır. Bu sistemler, enerji verimliliği ve performans açısından optimize edilir. PBFT ve Raft gibi algoritmalar, düşük enerji tüketimi ile yüksek işlem hızı sunar. Kapalı blok zincirler genellikle kurumsal uygulamalar için kullanılır, güvenlikten çok işlem verimliliği ön plandadır (Sharma and Lal 2020), (Anon 2019a), (Anon 2019b).

Hibrit blok zincirler, hem açık hem de kapalı blok zincir özelliklerini birleştirir. PoS, DPoS ve PoC gibi algoritmalar, güvenlik ve performans arasında denge kurar. Hibrit sistemler, farklı kullanım alanları için esneklik sağlarken, halka açık ve özel erişim bölümlerini bir arada kullanır (Anon 2019b).

Konsorsiyum blok zincirleri, bir grup kurum veya kuruluş tarafından yönetilen yarı kapalı sistemlerdir. PBFT, Paxos ve Tendermint gibi algoritmalar, bu blok zincirlerde yüksek performans ve güvenlik sunar. Bu algoritmalar, güvenilir iş ortakları arasında hızlı işlem doğrulama ve düşük enerji tüketimi sağladığı için özellikle kurumsal ve dağıtık sistemlerde tercih edilir (Anon 2019a), (Anon 2019b).

3.5. Nesnelerin interneti sistemleri (IoT)

IoT, fiziksel nesnelerin sensörler ve iletişim teknolojileriyle donatılarak veri toplama, analiz etme ve paylaşma yeteneğine sahip olmasıyla ortaya çıkan bir teknolojidir. IoT sistemleri; akıllı evler, akıllı şehirler, endüstriyel üretim sistemleri ve sağlık uygulamaları gibi çok çeşitli alanlarda uygulanmaktadır (Lu et al. 2020), (Hu et al. 2022). IoT sistemlerinin temel özelliklerinden *IoT Sistemlerinin Katmanlı Mimarisi,* IoT sistemleri genellikle üç temel katman üzerine inşa edilir: (Xing 2020) , (Sharma, Sangwan, and Singh 2023), (Çavdar, Tuğrul and Ercüment Öztürk 2018).



Şekil 3. 5*:* Nesnelerin interneti temel katmanları.

Algılama (Sensör) Katmanı; Sensörler ve fiziksel cihazlar, ortamdan veri toplar. Ağ ve İletişim Katmanı; Algılanan verilerin ağlar aracılığıyla iletildiği katmandır. Uygulama Katmanı; Verilerin işlendiği, analiz edildiği ve kullanıcıların eriştiği katmandır. Son yıllarda, uç bilişim ve sis bilişim gibi ara katmanların eklenmesiyle, veri işleme ve depolama süreçleri daha verimli ve güvenli hale getirilmiştir (Hu et al. 2022), (Skaria et al. 2024). *Güvenlik ve Gizlilik Sorunları,* IoT sistemlerinde verilerin büyük hacimli ve hassas olması, bu sistemleri saldırılara karşı savunmasız hale getirmektedir. Yetkisiz erişim, veri manipülasyonu, cihaz kimlik doğrulama eksikliği ve kişisel verilerin korunması gibi sorunlar sıklıkla karşılaşılmaktadır (Commey et al. 2024). *IoT ve Görüntü Aktarımı,* görüntü aktarımı, IoT sistemlerinin önemli bir bileşenidi. Özellikle güvenlik kameraları, sağlık sistemlerindeki görüntüleme cihazları ve akıllı şehir uygulamalarında, görüntü verilerinin güvenilir şekilde aktarılması kritik öneme sahiptir. Ancak mevcut IoT sistemleri, görüntü aktarımında güvenlik açıkları ve performans sorunlarıyla karşılaşmaktadır (Hu et al. 2022), (Skaria et al. 2024). *IoT'de Veri Yönetimi*, IoT sistemleri büyük miktarda veriyi işlemek ve yönetmek durumundadır. Blok zincir teknolojisi ile entegre edilen IoT sistemlerinde, verilerin güvenli şekilde depolanması ve erişim kontrolü önemli avantajlar sağlamaktadır. Bu entegrasyon, verilerin yetkisiz erişime karşı korunmasını ve veri bütünlüğünün sağlanmasını kolaylaştırmaktadır (Lu et al. 2020), (Tran et al. 2021).

DÖRDÜNCÜ BÖLÜM

YÖNTEM VE UYGULAMA

4.1. Donanım ve yazılım ortamı

Geliştirilen sistemin sağlıklı bir şekilde çalışabilmesi için donanım kadar yazılım uyumluluğu da kritik öneme sahiptir. Bu doğrultuda, hem merkezi kontrol birimi olan dizüstü bilgisayar hem de gömülü sistemler üzerinde Ubuntu 20.04 LTS işletim sistemi tercih edilmiştir. Robotik uygulamaların yönetimi ve haberleşmesi için ise Robot Operating System (ROS)'un Noetic sürümü kullanılmıştır. Kullanılan donanım bileşenleri ve sistemin fiziksel altyapısı Tablo 4.1'de özetlenmektedir.

Tablo 4. 1: Donanım bileşenleri ve teknik özellikler.

Donanım Bileşeni	**Teknik Özellikler**
Geliştirme Kartı	Raspberry Pi 3 Model B – 1.2 GHz 64-bit quad-core ARM Cortex-A53, 1 GB RAM
Robotik Platform	TurtleBot3 Burger – OpenCR kontrol kartı (STM32F7), DC motorlar, LDS-01 LiDAR, IMU, entegre Raspberry Pi 3 Model B, ROS uyumu
Kamera 1	Logitech Brio 100 – 1080p/30fps, 2MP CMOS sensör, 58° görüş açısı, sabit odak, dahili mikrofon, USB-A bağlantı, gizlilik kapağı
Kamera 2	Everest SC-802 – 1.3 MP CMOS sensör, 1280x960 çözünürlük, 10–15 FPS, LED aydınlatma, sabit odak, dahili mikrofon
Dizüstü Bilgisayar	Lenovo ThinkPad T460 – Intel Core i5-6200U, 8 GB DDR3L RAM, 240 GB SSD, 14" FHD ekran, Intel HD Graphics 520
İletişim Modülü	Dahili Wi-Fi (IEEE 802.11n) ve Bluetooth 4.1
Güç Kaynağı	12V Li-ion batarya (TurtleBot3), 5V 2.5A adaptör (Raspberry Pi için), 6-cell batarya (ThinkPad)

ROS Noetic, Ubuntu 20.04 ile tam uyumlu olup, TurtleBot3 Burger platformu ile entegre çalışabilmektedir. Ayrıca, blok zincir ağına veri aktarımı için Python 3 tabanlı kütüphaneler (özellikle web3.py) ve görüntü işleme işlemleri için OpenCV gibi ek yazılımlar kurulmuştur. IoT cihazlarından elde edilen veriler, güvenli, merkeziyetsiz ve değiştirilemez bir biçimde blok zincir ağına aktarılmasını hedeflemektedir. Sistem mimarisi bu amaç doğrultusunda belirli yazılım bileşenleri ve platformlar etrafında

şekillendirilmiştir. Kullanılan temel teknolojiler ve tercih edilme nedenleri aşağıda Tablo 4.2' de sunulmaktadır.

Tablo 4. 2: Yazılım bileşenleri ve kullanım nedenleri

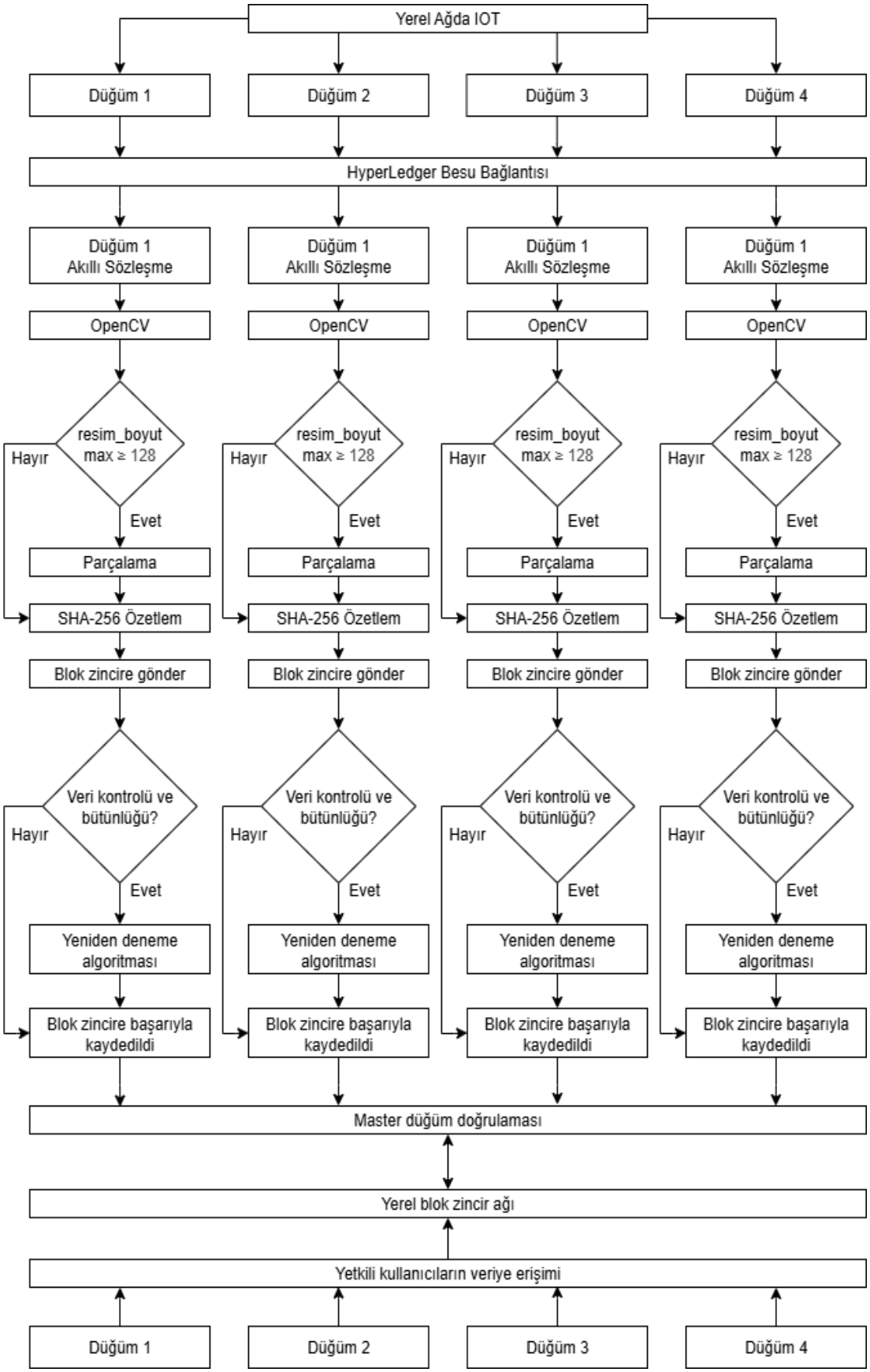
Yazılım Bileşeni	**Kullanım Nedeni**
Python SDK	IoT cihazlarından (örneğin Raspberry Pi) gelen görüntülerin toplanmasını sağlamak için özel olarak geliştirilmiştir. Web3.py kütüphanesi ile blok zincir bağlantısı gerçekleştirilmiştir.
Web3.py	Python ortamından Ethereum uyumlu blok zincir ağı ile etkileşim kurmak için tercih edilmiştir. Güvenli, imzalı işlem gönderimi sağlar.
Solidity	Akıllı sözleşme yazımı için kullanılan temel programlama dilidir. Blok zincire veri kaydının şeffaf ve değiştirilemez biçimde gerçekleştirilmesini sağlamaktadır.
Hyperledger Besu	Ethereum uyumlu, izinli (permissioned) özel bir blok zincir ağı kurulabilmesini mümkün kılar. QBFT Mutabakat algoritması ile güvenli ve ölçeklenebilir bir ağ yapısı sağlanmıştır.

4.2. Sistem mimarisi

Geliştirilen sistem üç katmanlı bir mimari yapıya sahiptir. IoT Katmanı, device klasörü altında yer alan photo.py modülü aracılığıyla IoT cihazları üzerindeki kameralardan görüntü veri alınmaktadır. Veri Aktarım Katmanı (Python SDK), main.py dosyası aracılığıyla tetiklenen veri toplama süreci, contract.py ve deploy.py dosyaları ile Web3.py üzerinden blok zincir ağına veri aktarımını gerçekleştirmektedir. Akıllı Sözleşme, contract.sol içerisinde tanımlı olan storeData, getData, deleteData gibi fonksiyonlar kullanılarak verilerin blok zincir ağına yazılması ve erişimi sağlanmaktadır.

Önerilen sistemin tasarımı ve veri işleme mekanizması aşağıdaki Şekil 4.1' de gösterilmiştir. Bu tasarım IoT cihazlarından alınan görüntü verilerinin Hyperledger Besu tabanlı blok zincir ağına dağıtık şekilde aktarılmasını, zincir üzerinde güvenli ve bütünlük garantisiyle kaydedilmesini sağlar. Her bir IoT düğümü, OpenCV kütüphanesi ile elde edilen görüntüleri öncelikle boyut kontrolünden geçirerek, gerekli durumlarda parçalara ayırmakta ve her parçaya SHA-256 algoritması ile bütünlük doğrulama işlemi uygulamaktadır. Ardından, veriler blok zincire gönderilmekte; bütünlük ve eksiklik denetimi sağlandıktan sonra yeniden gönderim algoritması (retry)

gerektiğinde devreye girmektedir. Tüm veri güvenli bir şekilde kayıt altına alındığında, master node doğrulama adımı gerçekleştirilir ve yalnızca yetkilendirilmiş kullanıcıların erişebileceği şekilde yerel blok zincir ağına aktarım tamamlanır. Bu veri aktarım süreci, TurtleBot ve Raspberry Pi cihazlarından gelen görüntülerin blok zincir ağına iletilmesini ve bu verilerin bütünlük/doğruluk denetimini kapsamaktadır. Söz konusu veri iletim akışı aşağıdaki adımlar doğrultusunda yapılandırılmıştır.



Şekil 4. 1 Sistemin mimari yapısı.

4.3. Geliştirme süreci

Sistemin genel çalışması aşağıdaki şekilde özetlenebilir:

- Kullanıcı main.py üzerinden süreci başlatır.
- Python SDK, device/ klasöründeki ilgili modüller ile veri toplar.
- Toplanan veri, storeData fonksiyonu kullanılarak contract.sol dosyasındaki akıllı sözleşmeye gönderilir.
- Sözleşme, gelen veriyi msg.sender adresine göre blok zincir ağına kaydeder.
- read.py modülü üzerinden veriler sorgulanabilir. Ayrıca, get_chunk_length ve get_chunk_missing gibi fonksiyonlar ile eksik veri kontrolü yapılabilir.

4.3.1. Akıllı sözleşme yapısı

Geliştirilen akıllı sözleşmede, her bir kullanıcıya ait çoklu veri kaydının saklanabilmesi için aşağıdaki Algoritma 4.1' deki yapı kullanılmıştır.

Algoritma 4.1: Akıllı Sözleşme Temel Yapısı.

```
1:  struct Record {
2:      string dataType;
3:      string dataÖzetleme;
4:      string metadata;
5:      uint timestamp;
6:  }
7:  mapping(address => Record[]) public records;
```

Algoritma 4.1, akıllı sözleşme tabanlı veri yönetim sisteminin temel veri yapısını tanımlamak üzere geliştirilmiştir. Sistem, IoT cihazları tarafından toplanan çoklu veri türlerinin blok zincir üzerinde düzenli ve erişilebilir bir şekilde saklanmasını amaçlamaktadır. Algoritmada yer alan struct Record yapısı, blok zincirde saklanacak veri nesnelerini tanımlayan standart bir veri modelidir. Bu yapı sayesinde veri türü, içerik özeti, açıklayıcı bilgiler ve zaman bilgisi bir bütün halinde tutulur. dataType alanı, kaydedilen verinin türünü belirtmekte olup ses, video, konum bilgisi veya görsel gibi farklı veri kategorilerinin sistemde ayrıştırılmasını sağlar. dataÖzetleme alanı, kaydın hash veya özet bilgisini içererek veri bütünlüğünün doğrulanmasını mümkün kılar. Böylece büyük hacimli verilerin doğrudan blok zincire kaydedilmesi yerine, yalnızca özet değerleri saklanarak performans artırılır ve veri gizliliği korunur. metadata alanı, veriye ilişkin açıklayıcı ek bilgiler içerir ve verinin hangi cihazdan

alındığı ya da hangi uygulama kapsamında üretildiği gibi detayların yönetimini kolaylaştırır. timestamp alanı ise verinin blok zincirine kaydedildiği zaman bilgisini içerir ve zaman temelli analizler veya sıralı işlemler için referans sağlar. Son olarak, mapping(address => Record[]) ifadesi sayesinde her kullanıcı adresi için birden fazla veri kaydı tutulabilir. Bu yapı, farklı cihazlardan veya kullanıcı hesaplarından gelen verilerin ayrı ayrı organize edilmesine olanak tanır. Ayrıca public erişim belirleyicisi, bu verilere ağ üzerindeki herkesin erişebilmesini, ancak yalnızca yetkili işlemlerle güncellenebilmesini sağlar. Bu yapı ile veri türü, veri özetleme değeri, açıklayıcı metadata ve zaman damgası ile birlikte kullanıcıya özel kayıt tutulmakta, böylece veri bütünlüğü ve izlenebilirlik sağlanmaktadır.

4.3.2. Örnek kod kullanımı

Aşağıda, görsel bir verinin parça (chunk) yapısı içerisinde özetleme'lenerek blok zincire gönderilmesini sağlayan örnek bir yapı Algoritma 4.2' de gösterilmektedir.

Algoritma 4.2: Parçalama ve Özetlem Algoritma Temel Yapısı.

```
1:  Load image chunk data(chunk)
2:  function STOREIMAGECHUNK(chunk_data, metadata)
3:      Compute özetleme of image chunk:
4:      → data_özetleme ← sha256(chunk_data)
5:      Store özetleme and metadata on blockchain:
6:      → store_data(data_type = "image", data_özetleme = data_özetleme, metadata
= metadata)
7:  end function
8:  // Örnek Kullanım
9:  chunk_data ← "image_chunk_1"
10: metadata ← "chunk1_of_10"
11: STOREIMAGECHUNK(chunk_data, metadata)
```

Algoritma 4.2, büyük boyutlu görsel verilerin blok zincire verimli bir şekilde kaydedilebilmesi amacıyla tasarlanmıştır. Büyük veri nesnelerinin doğrudan blok zincir ağına yazılması, hem maliyet hem de performans açısından verimsiz olduğundan, bu algoritmada parçalama ve özetleme yaklaşımları kullanılmaktadır. Algoritma, ilk adımda bir görüntü parçasının belleğe yüklenmesiyle başlar. Bu parça, büyük bir görüntünün daha küçük veri bloklarına ayrılmasıyla elde edilmiştir. Böylece, sistem büyük veri kümelerini parça parça işleyerek yönetilebilir hale getirir. Ardından, *STOREIMAGECHUNK* isimli fonksiyon çağrılır ve bu fonksiyon, gelen parça üzerinde özetleme işlemi gerçekleştirir. Özetleme işlemi için SHA-256

algoritması kullanılmaktadır. SHA-256, kriptografik olarak güvenli bir özetleme algoritması olup, verinin bütünlüğünü sağlamak ve doğrulamak için tercih edilmiştir. Bu sayede, sisteme gönderilen her bir görüntü parçasının içeriği değişmeden korunabilir ve daha sonra doğrulanabilir hale gelir. Özetleme işleminin tamamlanmasının ardından, elde edilen özet değeri ve parçaya ilişkin meta veriler akıllı sözleşme üzerinden blok zincire kaydedilir. Bu aşamada, *store_data* fonksiyonu kullanılarak veri türü, özet değer ve meta veriler blok zincir ağına aktarılır. Bu yapı sayesinde, blok zincire yalnızca doğrulanabilir küçük boyutlu özet bilgileri kaydedilerek performans ve maliyet avantajı sağlanırken, sistem bütünlüğünden ödün verilmez. Algoritmanın sonunda verilen örnek kullanım, "image_chunk_1" isimli bir görüntü parçasının ve bu parçaya ait "chunk1_of_10" şeklindeki meta bilginin işlenmesini göstermektedir. Bu örnek, algoritmanın gerçek kullanım senaryolarına nasıl entegre edileceğini göstermek amacıyla sunulmuştur.

4.3.3. Konfigürasyon yönetimi

Sistemin kurulumu sırasında kullanılan temel konfigürasyon dosyaları; genesis.json, blok zincir ağının başlangıç ayarlarını (örneğin gas limit, zaman damgası, chain ID) tanımlar. *config.toml,* her bir Besu düğümün çalışma portu, log seviyesi ve ağ bağlantılarını içerir. *static-nodes.json,* ağda bulunan diğer düğüm adreslerini ve bağlantı noktalarını belirler.

4.4. Kullanılan algoritmalar ve akış diyagramları

Bu bölümde, çalışma kapsamında kullanılan algoritmalar ve bunların sistem üzerindeki işlevleri detaylandırılmaktadır. Önerilen sistemde veri aktarımının güvenli ve verimli bir şekilde sağlanması için birden fazla algoritma entegre edilmiştir.

4.4.1. Özetleme algoritması (SHA-256)

SHA-256 algoritması, blok zincir ağına eklenen verilerin manipülasyona karşı korunması ve veri bütünlüğünün sağlanması amacıyla kullanılmaktadır. Her veri parçası, SHA-256 algoritması kullanılarak özetlemekte ve get özetlem() fonksiyonu ile blok zincir üzerinde saklanmaktadır. Bu yöntem, verinin manipüle edilmesi halinde özetleme değerinin değişmesini sağlayarak veri bütünlüğünü korur.

Algoritma 4.3: SHA-256 özetlem algoritması pseudo kodu.
1: Start 2: Input data 3: Function SHA256_Özetlemeing(data): 4: özetleme_value ← SHA256(data) 5: Return özetleme_value 6: End Function 7: Output ← SHA256_Özetlemeing(data) 8: End

Algoritma 4.3, sistemin bütünlük kontrolünü sağlamak amacıyla kullanılan SHA-256 özetleme işleminin temel mantığını göstermektedir. Blok zincir tabanlı sistemlerde, verilerin doğruluğunu ve değişmezliğini garanti altına almak için özetleme algoritmalarından yararlanmak kritik bir gerekliliktir. Bu amaçla, algoritmada SHA-256 fonksiyonu kullanılmıştır. SHA-256, kriptografik olarak güvenli bir özetleme algoritması olup, değiştirilemez ve sabit uzunlukta 256 bitlik çıktı üretmesiyle öne çıkar. Algoritma, kullanıcıdan veya sistemden alınan ham veri girişini işlemekle başlar. Girdi verisi *SHA256_Özetlemeing* fonksiyonuna aktarılır. Bu fonksiyon, gelen veriyi SHA-256 algoritması ile işleyerek verinin benzersiz ve tekil bir özet değerini üretir. Bu özet değer, aynı girdiye her zaman aynı çıktıyı üretirken, en ufak bir değişiklikte tamamen farklı bir çıktı üretme özelliğine sahiptir. Bu özellik, veri doğrulama süreçlerinde yüksek güvenlik sağlar. Fonksiyon, ürettiği özet değeri sistemin diğer bileşenlerine veya blok zincire kayıt için geri döner. Son adımda, algoritma çıktısı olarak bu özet değer elde edilir ve süreç tamamlanır. Bu yapı sayesinde, büyük veri nesnelerinin doğrudan kaydedilmesi yerine, içeriklerinin dijital imzası niteliğinde özet bilgilerinin kaydedilmesi mümkün hale gelir. Böylece hem veri bütünlüğü korunur hem de sistemin işlem yükü azaltılır.

4.4.2. Veri parçalama algoritması

Parçalama algoritması, büyük boyutlu verilerin blok zincir ağına etkin ve güvenli şekilde aktarılmasını sağlar. appendChunkData() fonksiyonu, alınan görüntü/video verilerini küçük parçalara bölerek blok zincire ekler. Bu yöntem, ağ bant genişliğini optimize ederek büyük verilerin daha verimli aktarılmasını sağlar. Eksik chunk'lar, get chunk() fonksiyonu ile yeniden çağırılarak yalnızca eksik kısımların tekrar aktarılması sağlanmaktadır.

Algoritma 4.4: Parçalama algoritması pseudo kodu.

```
FUNCTION Chunking(data, chunk_size):
    IF chunk_size > MAX_CHUNK_SIZE THEN
        THROW Error('Chunk size too large')
    chunks ← empty list
    i ← 0
    WHILE i < LENGTH(data) DO
        chunk ← data[i : i + chunk_size]
        chunk_hash ← SHA256_hashing(chunk)
        chunks.append((chunk, chunk_hash))
        i ← i + chunk_size
    RETURN chunks
FUNCTION AppendChunkData(chunks):
    FOR each chunk IN chunks DO
        chunk_data, chunk_hash ← chunk
        SEND chunk_data TO blockchain WITH chunk_hash
```

Algoritma 4.4, büyük boyutlu verilerin yönetilebilir parçalara bölünerek işlenmesini sağlamak amacıyla geliştirilmiştir. Blok zincir sistemlerinde büyük veri yüklerinin doğrudan zincire kaydedilmesi hem maliyet hem de performans açısından önemli sınırlamalar doğurur. Bu nedenle, verinin parçalara ayrılarak işlenmesi ve her bir parçanın özet bilgisinin zincire kaydedilmesi daha verimli bir çözüm sunar. Algoritma, *Chunking* isimli fonksiyonla başlamakta ve iki temel girdi almaktadır: işlenecek ham veri (*data*) ve bölünecek parça boyutu (*chunk_size*). Fonksiyonun ilk aşamasında, belirlenen parça boyutunun sistemde tanımlanan maksimum sınırı aşıp aşmadığı kontrol edilmektedir. Bu kontrol, sistem kaynaklarının aşırı kullanımını önlemek ve performansı korumak amacıyla yapılır. Parça boyutu uygun değilse hata mesajı üretilerek işlem sonlandırılır. Uygun parça boyutu onaylandıktan sonra, algoritma parçalama işlemine geçer. Veri üzerinde belirli aralıklarla dolaşarak, belirtilen boyutta alt parçalar oluşturur. Her bir parçanın SHA-256 özet değeri hesaplanarak hem veri hem de özet değeri bir listeye eklenir. Bu süreç, tüm veri işlenene kadar devam eder. Bu yapı sayesinde, verinin hem içerik hem de bütünlük bilgileri birlikte yönetilir.

Algoritmanın ikinci aşamasında, *AppendChunkData* isimli fonksiyon devreye girer. Bu fonksiyon, daha önce oluşturulan tüm parça ve özet çiftlerini sırasıyla işler. Her bir parça ve ona ait özet değeri, blok zincire gönderilmek üzere hazırlanır. Böylece yalnızca verinin kendisi değil, doğrulanabilirlik için gerekli olan özet bilgisi de blok zincir üzerinde kayıt altına alınır. Sonuç olarak, bu algoritma büyük boyutlu veri yönetimini kolaylaştırmakta, sistem kaynaklarını daha verimli kullanmakta ve blok zincir üzerinde yüksek doğrulukta veri bütünlüğü sağlamaktadır.

4.4.3. RPC protokolü

Blok zincire veri aktarımı ve sorgulama işlemleri HTTP RPC protokolü kullanılarak gerçekleştirilmektedir. run() fonksiyonu, bu protokol aracılığıyla blok zincir ağına bağlanır ve veri ekleme/sorgulama işlemlerini yürütür. JSON-RPC protokolü, IoT cihazlarında kaynak tüketimini minimize ederek veri aktarımını hızlandırır.

Algoritma 4.5: RPC Algoritması Pseudo Kodu

```
FUNCTION RPC_Connect(node_url):
    connection ← CONNECT_TO_NODE(node_url)
    IF connection IS NOT ESTABLISHED THEN
        THROW Error('Connection Failed')
    RETURN connection
FUNCTION SendData(connection, data, signature):
    request ← {
        "method": "eth_sendTransaction",
        "params": [data, signature]
    }
    connection.send(request)
```

Algoritma 4.5, blok zincir ağı ile dış sistemlerin güvenli ve standart bir protokol üzerinden iletişim kurmasını sağlamak amacıyla geliştirilmiştir. Blok zincir mimarisinde dış sistemlerin ağa veri gönderebilmesi veya zincir üzerindeki işlemleri gerçekleştirebilmesi için genellikle Uzaktan Prosedür Çağrısı (RPC - Remote Procedure Call) protokolü kullanılmaktadır. Bu algoritma, söz konusu protokolün temel kullanım mantığını modellemektedir. Algoritma, *RPC_Connect* isimli fonksiyonla başlar. Bu fonksiyon, blok zincir düğümüne bağlantı kurmak için hedef düğümün URL adresini girdi olarak alır. Fonksiyon, belirtilen adrese bağlantı kurulmasını sağlar ve bağlantı durumunu kontrol eder. Eğer bağlantı başarılı bir şekilde kurulamazsa, sistem kullanıcıyı uyararak işlemi sonlandırır. Bu yapı, güvenilir bağlantı kontrolü sağlayarak yanlış yapılandırmalar veya bağlantı sorunları nedeniyle sistemin hata vermesinin önüne geçer. Bağlantının başarılı olması durumunda, bağlantı nesnesi sonraki işlemler için geri döndürülür. İkinci aşamada, *SendData* isimli fonksiyon devreye girer. Bu fonksiyon, blok zincir ağına gönderilmek üzere hazırlanmış veri ve bu veriye ait dijital imzayı parametre olarak alır. Fonksiyon, *eth_sendTransaction* yöntemi ile blok zincire işlem göndermek üzere bir RPC isteği hazırlar. Bu istek, Ethereum uyumlu ağlarda standart olarak kullanılan işlem gönderme yöntemidir. Hazırlanan istek, bağlantı üzerinden ağa iletilir. Bu yapı sayesinde, sistem dışındaki bileşenler (örneğin IoT cihazları veya kullanıcı arayüzleri), güvenli bir

bağlantı üzerinden blok zincir ağına doğrudan işlem gönderebilir. Bu da merkeziyetsiz uygulamaların gerçek zamanlı veri işleme ve zincirle etkileşim yeteneklerini güçlendirir.

4.4.4. PoA mutabakat algoritması

Hyperledger Besu platformunda kullanılan QBFT (IBFT 2.0) mutabakat algoritması, PoA (Proof of Authority) tabanlı bir yapıya sahiptir. Bu mekanizma, yalnızca yetkilendirilmiş düğümlerin blok üretme ve doğrulama işlevlerini gerçekleştirmesine izin vererek blok zincir ağının güvenliğini artırmaktadır. Algoritma 4.6, blok zincir ağlarında hızlı ve düşük maliyetli mutabakat sağlamak amacıyla kullanılan PoA mekanizmasının temel işleyişini modellemektedir. PoA, PoW algoritmalarından farklı olarak, yalnızca önceden yetkilendirilmiş düğümlerin blok üretme hakkına sahip olduğu, merkeziyetsizliği sınırlı ancak performansı yüksek bir mutabakat yaklaşımıdır.

Algoritma 4.6: PoA Algoritması Pseudo Kodu

```
1:  FUNCTION PoA_Consensus(block):
2:      IF block_producer IS AUTHORIZED THEN
3:          IF block IS VALID THEN
4:              ADD block TO blockchain
5:          ELSE:
6:              REJECT block (Invalid Data)
7:      ELSE:
8:          REJECT block (Unauthorized Producer)
4:          THROW Error('Connection Failed')
```

Algoritma, *PoA_Consensus* isimli fonksiyonla başlar ve ağa önerilen bir blok üzerinde işlem yapar. İlk olarak, bloğu üreten düğümün yetkili olup olmadığı kontrol edilir. Bu kontrol, ağın güvenilirliğini sağlamak için kritik öneme sahiptir çünkü yalnızca önceden belirlenen ve kimliği doğrulanmış düğümler blok üretebilir. Eğer düğüm yetkili değilse, blok doğrudan reddedilir ve işlem sonlandırılır. Yetkilendirme doğrulandıktan sonra, önerilen bloğun geçerliliği kontrol edilir. Bu aşamada, bloğun yapısal bütünlüğü, işlem doğruluğu ve ağ kurallarına uygunluğu değerlendirilir. Eğer blok geçerliyse, zincire eklenir ve sistemin sürekliliği sağlanır. Geçersiz bir blok ise ağ bütünlüğünü korumak amacıyla reddedilir. PoA algoritmasının bu yapısı, yüksek işlem hacmine sahip özel veya izinli blok zincir uygulamalarında performansı artırmak için tercih edilmektedir. Yetkilendirilmiş üreticiler sayesinde ağda işlem onay süreleri

düşerken, kaynak tüketimi de minimum seviyede tutulur. Bununla birlikte, algoritmanın merkeziyet derecesinin yüksek olması, tamamen açık ağlara kıyasla sınırlı merkeziyetsizlik sunmasına neden olmaktadır.

4.4.5. OpenCV algoritması (Görüntü ve video veri toplama)

Görüntüler photo.py dosyasında cv2.VideoCapture() fonksiyonu ile yakalanmakta ve cv2.imencode() fonksiyonu ile JPG formatına sıkıştırılarak daha verimli aktarım sağlanmaktadır. Benzer şekilde videolar video.py dosyasında cv2.VideoWriter() fonksiyonu ile MP4 formatında kaydedilmektedir. Algoritma 4.7, IoT tabanlı görüntü toplama sistemlerinde, bağlı kameralar üzerinden anlık görüntü elde etmek ve bu görüntüyü sıkıştırarak işlemeye hazır hale getirmek amacıyla tasarlanmıştır.

Algoritma 4.7: Görüntü yakalama algoritması pseudo kodu.

```
1:  FUNCTION CaptureImage():
2:      TRY:
3:          camera ← OPEN_CAMERA(0)
4:          image ← camera.read()
5:          compressed_image ← COMPRESS(image, 'JPG')
6:          RETURN compressed_image
7:      EXCEPT Error:
8:          RETRY CaptureImage() UNTIL MAX_RETRIES
```

Görüntü tabanlı uygulamalarda gerçek zamanlı veri toplama, sistem performansı ve işlem güvenilirliği açısından kritik öneme sahiptir. Algoritma *CaptureImage* isimli fonksiyonla başlar ve öncelikle sistemde tanımlı olan kameraya erişim sağlamaya çalışır. Kamera açma işlemi *OPEN_CAMERA(0)* komutu ile başlatılır. Bu adım, genellikle sistemdeki varsayılan veya belirtilen kamera donanımına erişim sağlar. Başarılı bir bağlantı kurulmasının ardından, kameradan bir kare alınarak *image* değişkenine aktarılır. Bu işlem, cihazın o anda algıladığı görsel verinin kaydedilmesini sağlar. Alınan ham görüntü, blok zincire doğrudan gönderilmeden önce *COMPRESS* fonksiyonu kullanılarak 'JPG' formatında sıkıştırılır. Görüntü sıkıştırma işlemi, veri boyutunu azaltarak hem depolama hem de iletim süreçlerinde bant genişliği ve kaynak kullanımını optimize eder. Böylece sistem, ağ üzerindeki veri trafiğini minimize ederken görsel bütünlüğü makul seviyede korur. Görüntü alma veya kamera

erişiminde hata oluşması durumunda algoritma, hata yönetimi mekanizmasını devreye alır. Hata oluştuğunda, tanımlı maksimum deneme sınırına kadar yeniden görüntü yakalama işlemi gerçekleştirilir. Bu yapı, geçici bağlantı veya donanım sorunlarına karşı sistemin dayanıklılığını artırır ve işlemin başarıyla tamamlanmasını garanti altına alır. Sonuç olarak bu algoritma, görüntü tabanlı veri toplama süreçlerinde güvenilirlik, performans ve veri yönetimi açısından optimize edilmiş bir çözüm sunmaktadır.

4.4.6. Yeniden deneme (Retry) algoritması

getchunk() ve getchunklength() fonksiyonları, eksik chunk'ların tespit edilmesini ve yalnızca bu eksik parçaların yeniden gönderilmesini sağlamaktadır. Bu algoritma, bant genişliğini koruyarak veri aktarımını optimize eder ve veri kaybını önler. Algoritma 4.8, blok zincir tabanlı veri aktarım süreçlerinde eksik iletilen veya kaybolan veri parçalarının tespit edilmesi ve yeniden gönderilmesi için geliştirilmiştir.

Algoritma 4.8: Yeniden deneme (retry) algoritması pseudo kodu.

```
FUNCTION DetectMissingChunks(total_chunks, received_chunks):
    missing_chunks ← []
    FOR i FROM 0 TO total_chunks - 1 DO
        IF i NOT IN received_chunks THEN
            missing_chunks.APPEND(i)
    RETURN missing_chunks
FUNCTION ResendMissingChunks(missing_chunks):
    FOR each index IN missing_chunks DO
        chunk ← get_chunk(index)
        length ← get_chunk_length(index)
        SEND chunk TO blockchain WITH length
```

Dağıtık sistemlerde ağ gecikmeleri, işlem hataları veya bağlantı sorunları nedeniyle veri iletiminde kayıplar yaşanabilmektedir. Bu algoritma, bu tür durumları tespit edip otomatik olarak iyileştirme sağlamak amacıyla kullanılmaktadır. Algoritmanın ilk aşaması *DetectMissingChunks* isimli fonksiyonla başlamaktadır. Bu fonksiyon, beklenen toplam parça sayısı (*total_chunks*) ve başarıyla alınan parçaların listesi (*received_chunks*) parametreleriyle çalışır. Döngü yardımıyla tüm parça indeksleri kontrol edilir ve alınmamış olanlar *missing_chunks* listesine eklenir. Bu yapı sayesinde, eksik kalan veri parçaları sistematik olarak tespit edilerek kayıt altına alınır. İkinci aşama olan *ResendMissingChunks* fonksiyonu ise tespit edilen eksik parçaların yeniden gönderilmesini sağlar. Bu aşamada her eksik parça, ilgili indeks üzerinden sistemden alınır ve boyut bilgisiyle birlikte tekrar gönderilmek üzere hazırlanır.

Yeniden gönderme işlemi sırasında hem parça verisi hem de boyut bilgisi blok zincire iletilir. Bu, parçaların doğruluğunu ve bütünlüğünü koruyarak eksik iletimlerin tamamlanmasını sağlar. Bu algoritma, blok zincir uygulamalarında veri kaybı riskini en aza indirirken, veri bütünlüğü ve işlem güvenilirliği açısından sistemin dayanıklılığını artırmaktadır. Ayrıca, manuel müdahale gerektirmeden otomatik hata iyileştirme sağladığı için kullanıcı deneyimini iyileştirir ve sistemin sürdürülebilirliğine katkıda bulunur.

4.5. Kodların genel akışı

Geliştirilen sistemin yazılım akışı, modüler bir yapı ile organize edilmiştir. Sistemin başlatılması, main.py üzerinden yapılmakta; bu dosya aracılığıyla toplanan veriler zincire aktarılmaktadır. Bu süreçte *main.py,* kullanıcı tetiklemesiyle çalışır, cihaz modüllerini çağırır. *photo.py,* görüntü verilerini toplar. *contract.py,* toplanan veriyi web3.py aracılığıyla blok zincir ağına gönderir. *contract.sol,* gelen verileri, kullanıcı adresine göre zincire kaydeder. *read.py,* zincirden veri okuma ve eksik parça kontrol işlemlerini yürütür. Bu yapı sayesinde sistem, uçtan uca güvenli ve parçalı veri aktarımına olanak tanımaktadır.

4.6. Veri bütünlüğü ve parçalama kontrol mekanizması

Parçalı veri aktarımında her bir veri, tanımlı boyutta bölünerek zincire gönderilmektedir. Sistemde get_chunk_missing() fonksiyonu ile eksik parça kontrolü yapılmaktadır. Bu mekanizma sayesinde herhangi bir parçanın eksik olması durumunda sistem, eksik olan chunk_id'leri belirleyerek yeniden gönderim yapılmasına imkân tanır. Algoritma 4.8, dağıtık veri aktarım süreçlerinde eksik kalan veya gönderilemeyen veri parçalarının otomatik olarak tespit edilmesini sağlamak amacıyla tasarlanmıştır.

Algoritma 4.8: Gönderilemeyeni tekrar göndermek için pseudo kodu.

```
FUNCTION GetMissingChunks(expected_total, existing_chunks):
    missing ← []
    FOR i FROM 0 TO expected_total - 1 DO
        IF i NOT IN existing_chunks THEN
            missing.APPEND(i)
    RETURN missing
```

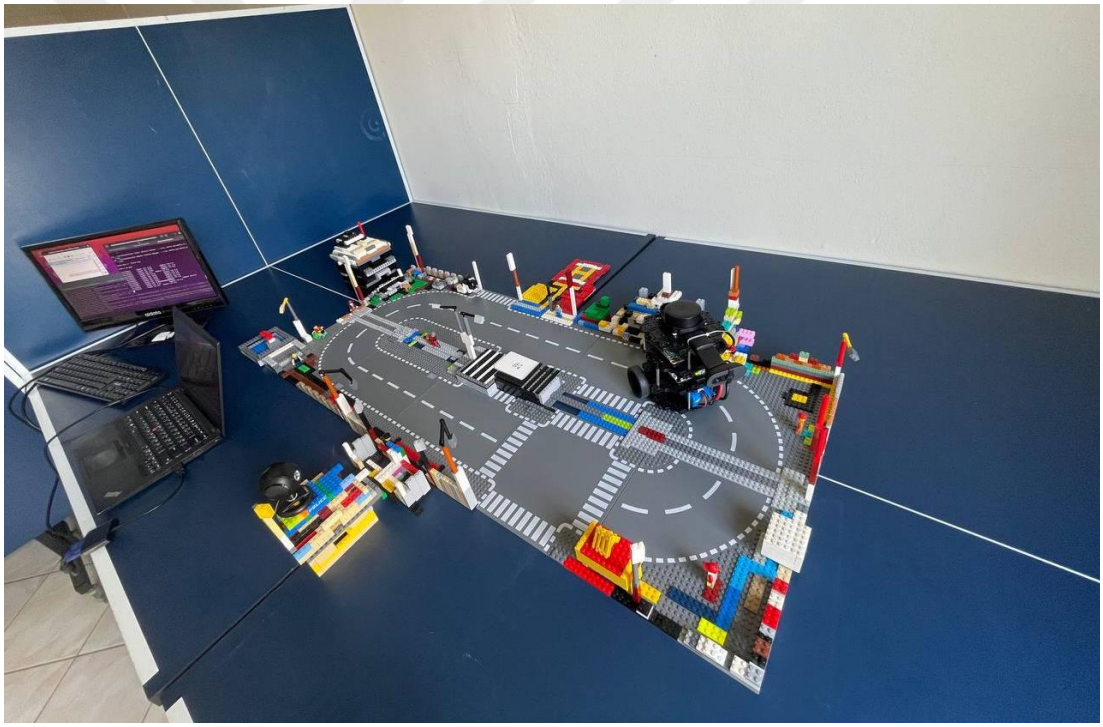
Örnek kullanım: missing = get_chunk_missing(expected=10, existing=[0,1,2,3,5,6]) Çıktı: [4, 7, 8, 9] Bu kontrol, bütünlük (integrity) açısından kritik öneme sahiptir.

Özellikle parça tabanlı veri iletimlerinde, ağ hataları veya işlem kesintileri nedeniyle bazı parçaların hedef sisteme ulaşamaması yaygın bir durumdur. Bu algoritma, bu tür durumların yönetimini kolaylaştırmakta ve eksik parçaların yeniden gönderilmesi için gerekli olan ilk adımı oluşturmaktadır. Algoritma *GetMissingChunks* fonksiyonu ile başlamakta ve iki temel parametre almaktadır. Bunlardan ilki, beklenen toplam parça sayısını belirten *expected_total* değeridir. İkinci parametre ise başarıyla alınmış olan parçaların listesi olan *existing_chunks* dizisidir. Algoritma, 0'dan başlayarak tüm beklenen parçaların indekslerini sırayla kontrol eder. Eğer mevcut parça listesinde yer almayan bir indeks tespit edilirse, bu indeks *missing* listesine eklenir. Döngü tamamlandığında, eksik parçaların indekslerini içeren liste geri döndürülür. Bu liste, sistemin eksik parçaları yeniden gönderebilmesi için temel girdi olarak kullanılabilir. Böylece veri bütünlüğü korunur ve iletim sürecinde oluşabilecek kayıplar minimuma indirilmiş olur. Bu yaklaşım, özellikle parçalı veri iletiminin yoğun olduğu blok zincir, IoT veya büyük veri uygulamalarında kritik öneme sahiptir. Eksik verilerin tespiti ve yeniden gönderim sürecinin otomatikleştirilmesi, sistemin güvenilirliğini ve performansını önemli ölçüde artırmaktadır.

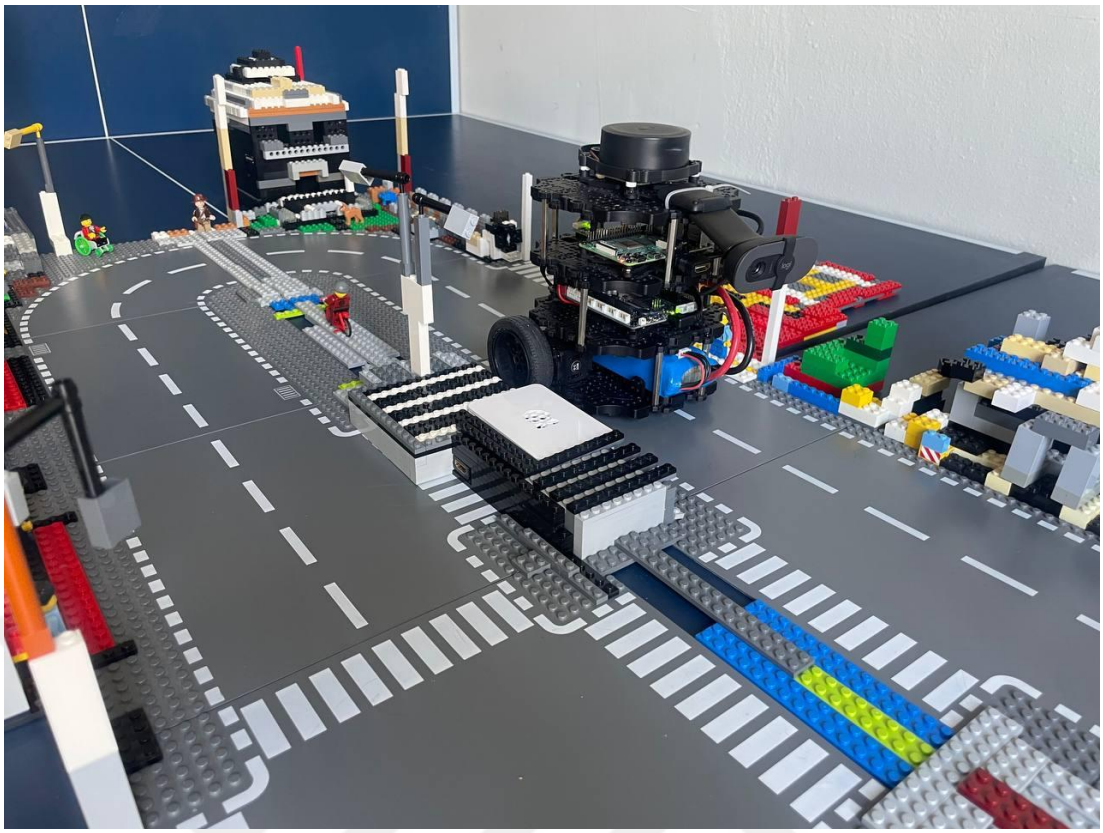
BEŞİNCİ BÖLÜM

ÖNERİLEN BLOK ZİNCİR TABANLI SİSTEMİN TASARIMI

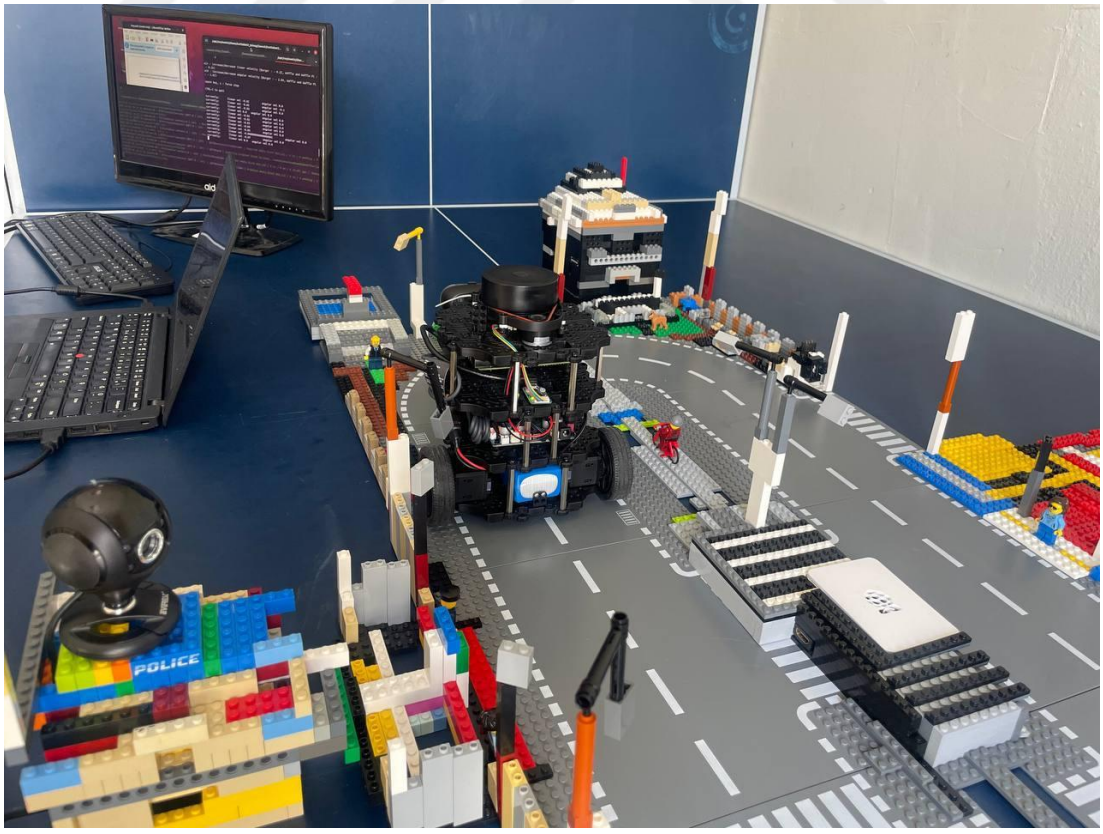
IoT sistemlerinde görüntü aktarımının güvenli ve verimli şekilde gerçekleştirilmesi için geliştirilen önerilen sistemimiz, Hyperledger Besu altyapısı üzerinde çalışmakta ve 4 düğümden oluşan dağıtık bir mimari kullanmaktadır. Şekil 5.1, Geliştirilen sistemin TurtleBot3 ve Raspberry Pi bileşenleriyle oluşturulmuş test ortamındaki fiziksel kurulumunu göstermektedir. LEGO bloklarından oluşturulan mini şehir simülasyon alanında, robotun görüntü toplama, veri bütünlüğü kontrolüve blok zincire güvenli aktarım işlemleri başarıyla test edilmiştir. Sistem, Hyperledger Besu tabanlı 4 düğümlü özel blok zincir ağına HTTP RPC üzerinden bağlanmakta ve verileri parçalayarak aktarım sağlamaktadır.



Şekil 5. 1 Test ortamı veri toplama ve blok zincire aktarım simülasyonu.



Şekil 5. 2 Test ortamı veri toplama ve blok zincire aktarım simülasyonu.



Şekil 5. 3 Test ortamı veri toplama ve blok zincire aktarım simülasyonu.

5.1. Sistemin çalışma prensibi

Önerilen sistem mimarisi, Şekil 1.1 ve Şekil 4.1'de detaylı olarak gösterildiği üzere, IoT cihazları aracılığıyla toplanan görsel verilerin bütünlüğü korunarak blok zincire kaydedilmesini sağlamaktadır. Bu süreçte TurtleBot ve Raspberry Pi cihazları, yerel ağ üzerinden Hyperledger Besu blok zincir ağına bağlı düğümler aracılığıyla işlem gerçekleştirmektedir. Sistemin omurgasında Hyperledger Besu'nun QBFT (IBFT 2.0) mutabakat mekanizması yer almakta olup, bu mekanizma sayesinde dağıtık düğümler arasında veri bütünlüğü ve güvenilirliği sağlanmaktadır. Şekil 1.1'de ise sistemin genel işlem akışı gösterilmektedir. Görüldüğü üzere süreç, IoT cihazlarının ağa bağlantısı ve blok zincir bağlantısının sağlanmasıyla başlamakta, ardından görüntü işleme ve parçalama işlemleri gerçekleştirilmektedir. Veriler blok zincire kaydedildikten sonra, master düğüm doğrulama süreci devreye girmekte ve doğrulama tamamlandığında yetkilendirilmiş kullanıcıların veriye erişimi sağlanmaktadır. Şekil 4.1'de görüldüğü gibi, sistemde yer alan tüm düğümler, yerel IoT ağına bağlı olarak Hyperledger Besu ağına erişmekte ve kendilerine atanmış akıllı sözleşmeler üzerinden görüntü işleme sürecini başlatmaktadır. Bu süreçte OpenCV kütüphanesi kullanılarak elde edilen görüntülerin boyutu kontrol edilmekte, belirlenen maksimum sınırın aşılması durumunda *parçalama* işlemi uygulanarak veriler küçük parçalara bölünmektedir. Her parça, SHA-256 algoritması kullanılarak özetlenmekte ve özet değerleriyle birlikte blok zincire gönderilmektedir. Gönderim sonrasında veri bütünlüğü ve doğruluğu kontrol edilmekte, eksik parçalar tespit edilirse yeniden gönderim mekanizması devreye alınmaktadır. Bu sayede her bir parça başarılı şekilde blok zincire kaydedilene kadar işlem tekrarlanmaktadır. Tüm bu işlemler sırasında HTTP RPC protokolü kullanılarak veri gönderimi yapılmakta, startData() fonksiyonu ile veri aktarım süreci başlatılmakta ve appendChunkData() fonksiyonu ile parçalar blok zincire eklenmektedir. Blok zincire kaydedilen her parça SHA-256 özet değeri ile bütünlüğü korunarak depolanmaktadır. Verilerin daha sonra blok zincirden geri çağrılması durumunda getChunkData() fonksiyonu aracılığıyla özetleme doğrulaması yapılmakta ve olası veri manipülasyonları tespit edilmektedir. Bu bütünleşik yapı sayesinde, sistem hem veri güvenliğini hem de bütünlüğünü garanti altına alarak yüksek güvenilirlikte bir çözüm sunmaktadır.

5.2. Sistem tasarım adımları

Bu çalışma kapsamında geliştirilen sistem, görüntü verisinin uçtan uca güvenli ve verimli bir şekilde işlenmesini hedeflemektedir. Sistem tasarımı; görüntülerin IoT cihazları aracılığıyla alınması, büyük boyutlu verilerin 128 KB'lik parçalara bölünmesi, her parçanın SHA-256 algoritması ile özetlenmesi ve bu özet bilgilerinin meta verilerle birlikte Hyperledger Besu tabanlı blok zincir ağına kaydedilmesi adımlarını içermektedir. Ayrıca, master düğüm tarafından gerçekleştirilen bütünlük doğrulaması ile yalnızca eksik veya bozulmuş parçaların yeniden gönderilmesi sağlanarak ağ yükü azaltılmaktadır. Python tabanlı akıllı sözleşmeler ile yalnızca yetkilendirilmiş kullanıcıların veri erişimi mümkün kılınmakta, böylece hem veri güvenliği sağlanmakta hem de iletim verimliliği maksimuma çıkarılmaktadır. Önerilen sistemde görüntü alımı, TurtleBot, Raspberry Pi veya PC kameraları gibi IoT cihazları aracılığıyla gerçekleştirilmekte olup, bu cihazlardan anlık veya önceden kaydedilmiş görüntü verileri elde edilmektedir. Alınan büyük boyutlu görüntüler, daha verimli aktarım ve yönetim sağlamak amacıyla 128 KB'lik parçalara bölünmektedir. Bu yöntem sayesinde büyük verilerin yalnızca eksik kalan kısımları yeniden gönderilerek ağ üzerindeki yük azaltılmaktadır. Parçalama işlemi sonrası, her bir parça için SHA-256 algoritması kullanılarak özetleme değeri hesaplanmakta ve bu özetleme değeri, veri bütünlüğünün korunması için benzersiz bir kimlik işlevi görmektedir. Elde edilen özet değerleri ile birlikte dosya adı ve parça kimliği gibi meta veriler, Hyperledger Besu'nun IBFT 2.0 mutabakat algoritması kullanılarak blok zincire kaydedilmektedir. Blok zincire kaydedilen veriler daha sonra Master düğüm tarafından doğrulanmakta, özetleme değerleri karşılaştırılarak eksik veya değiştirilmiş parçalar tespit edilmektedir. Bu durumda yalnızca ilgili parçanın yeniden gönderimi sağlanarak veri bütünlüğü korunmaktadır. Son aşamada ise Python tabanlı akıllı sözleşmeler üzerinden yalnızca yetkilendirilmiş kullanıcıların verilere erişimi mümkün kılınmakta, böylece hem güvenlik sağlanmakta hem de yalnızca gerekli parçaların işlenmesiyle ağ verimliliği optimize edilmektedir.

5.3. Düğüm yapısı ve rolleri

Önerilen sistem mimarisi, dört farklı düğümden oluşmaktadır. Bu yapı, görüntü verilerinin güvenli ve kesintisiz şekilde aktarımını sağlamak amacıyla tasarlanmış olup, her node' un belirli sorumlulukları bulunmaktadır.

5.3.1. Düğüm 1 / Node 1 (master node - pc üzerinde)

a) *Blok zincir ağının yönetimi:* Master node, blok zincir ağını yöneterek verileri güvende tutar. Yeni blokların eklenmesi, düğümler arasındaki iletişim ve Mutabakat algoritmasını kontrol etme gibi temel görevleri yürütür.
b) *Yetkilendirme ve erişim kontrolü:* Kullanıcı doğrulama işlemlerini gerçekleştirir ve verilere erişim yetkisi olan cihazları belirler. Bu işlem, güvenli erişim sistemlerinin iyi çalışmasını sağlar.

5.3.2. Düğüm 2 / Node 2 (pc üzerinde)

a) *Ağ yükü dengeleme:* Ağda veri trafiğinin etkili bir şekilde aktarılmasını sağlar. Yük dengeleme mekanizması, Internet of Things (IoT) cihazlarının oluşturduğu devasa veri akışının daha istikrarlı bir şekilde yürütülmesine yardımcı olur.
b) *Veri akışı yönetimi:* Çok fazla trafik olduğunda paket kayıplarını azaltarak veri bütünlüğünü korur.

5.3.3. Düğüm 3 / Node 3 (turtlebot)

a) *Görüntü verisi toplama:* TurtleBot'un entegre kamerası, blok zincir ağına veri toplamak ve blok zincir ağına aktarmak için kullanılır.
b) *Hareket halinde veri toplama:* TurtleBot'un mobil yapısı, hareket halindeyken çevresel verileri sürekli olarak toplayabilir ve bu verilerin özetleme değerleri blok zincir ağına kaydedilir. Özellikle dinamik ortamlarda görüntü aktarımının sürekliliğini garanti etmek için bu yöntem çok önemlidir.

5.3.4. Düğüm 4 / Node 4 (raspberry pi)

a) *Raspberry pi kamerası ile görüntü toplama:* Raspberry Pi platformu, düşük maliyetli ve enerji verimli bir cihaz olması sebebiyle IoT uygulamalarında sıkça tercih edilmektedir. Raspberry Pi entegre kamerası aracılığıyla çevresel görüntü verileri toplanır ve blok zincir ağına kaydedilir.
b) *Enerji verimliliği:* Raspberry Pi'nin düşük enerji tüketimi, uzun süreli çalışmalarda avantaj sağlar ve IoT cihazlarının saha ortamlarında kesintisiz çalışmasına katkıda bulunur.

5.4. Hyperledger Besu mimari özellikleri

Özellikle eşler arası enerji ticareti sistemlerinde blok zinciri tabanlı uygulamalar için, kurumsal düzeyde bir Ethereum istemcisi olan Hyperledger Besu , hem izinli hem de izinsiz ağlarda çalışabilme kapasitesi nedeniyle çeşitli kullanım durumları için uygundur (Kim et al. 2022b) . Gecikme ve artan verimlilik açısından Besu'nun İstanbul Bizans Hata Toleransı (IBFT) 2.0 Mutabakat algoritmasının uygulanması, rakip blok zinciri çerçevelerinden daha iyi performans göstermiştir (Abdella et al. 2021) , (Pradhan et al. 2022). Performans testlerinde Ethereum ve Hyperledger Fabric'in RAFT'ına kıyasla kat kat daha yüksek verim ve 5 kat daha düşük gecikme süresi (Pradhan et al. 2022). Geleneksel genel blok zincirlerine kıyasla daha iyi ölçeklenebilirlik, güvenlik ve verimlilik sağlayan özellikleri, Hyperledger Besu'yu kurumsal blok zinciri çözümleri için arzu edilen bir seçim haline getiriyor (Capocasale, Gotta, and Perboli 2023) ; (Abdella et al. 2021) , (Leng et al. 2022).

Hyperledger Besu'nun öne çıkan özellikleri şunlardır:(Anon n.d.-a, Besu.Hyperledger)

Modüler mimari, Hyperledger Besu'nun esnek yapısı ve modüler mimarisi, onu çeşitli ağ türlerine kolayca entegre etme yeteneğine sahiptir. Bu özellik, özelleştirilmiş blok zincir çözümlerinin oluşturulmasında büyük bir avantaj sağlar (Tran et al. 2021).
Gelişmiş güvenlik, private transaction manager (PTM) gibi güvenlik araçları, Hyperledger Besu'nun veri gizliliğini ve bütünlüğünü destekler. PTM, blok zincir ağında belirli verilerin yalnızca yetkilendirilmiş kullanıcılar tarafından görülmesine izin verir (Skaria et al. 2024).
Ethereum sanal makinesi (EVM) desteği, EVM uyumluluğu, Hyperledger Besu'nun Ethereum tabanlı akıllı sözleşmelerin sorunsuz çalışmasını sağlar. Bu özellik sayesinde mevcut Ethereum altyapısıyla çalışan uygulamalar kolayca Hyperledger Besu'ya taşınabilir (Commey et al. 2024).

5.4.1. IBFT 2.0 mutabakat algoritması

IBFT 2.0 olarak da bilinen Istanbul Byzantine Fault Tolerance (IBFT) mutabakat algoritması, önerilen sistemin temelini oluşturur. Hyperledger Besu'da etkili ve güvenli bir algoritma olan IBFT 2.0, izinli (permissioned) ağlarda tercih edilen bir algoritmadır. Düşük gecikme süreleri ve yüksek işlem doğruluk oranı, IBFT 2.0'ın temel özellikleridir (Skaria et al. 2024).

5.4.2. IBFT 2.0 algoritmasının işleyişi

IBFT 2.0 algoritmasının işleyişi, üç temel bileşen üzerinden gerçekleştirilmektedir:

Proposer (Önerici), her blok oluşturma aşamasında, sistemdeki bir düğüm önerici olarak kabul edilir ve yeni bir blok oluşturma görevi verilir. Blok oluşturulduktan sonra, oluşturucu diğer doğrulayıcı noktalara iletilir (Commey et al. 2024).

Validator (Doğrulayıcı), doğrulayıcı düğümler, öncünün blokunu kontrol eder. IBFT 2.0 Mutabakat algoritmasına göre, ağdaki doğrulayıcı düğümlerin en az %66'sı (2/3) bloğun doğruluğunu onayladığında blok zincire dahil edilir (Tran et al. 2021).

IBFT 2.0'ın "Round Change" mekanizması, blok üretme sırasında önerici düğüm başarısız olduğunda hata toleransı sağlar. Blok üretilemezse, sistem yeni bir önerici seçer ve işlem devam eder. Bu yöntem, ağın sürekliliğini ve verileri korur (Martina et al. 2023).

5.4.3. IBFT 2.0'ın avantajları

IBFT 2.0 algoritması, aşağıdaki avantajlarıyla IoT sistemleri gibi zaman duyarlı uygulamalarda güvenli ve verimli çalışmaktadır. Düşük gecikme süresiyle yüksek işlem hızı sağlar, bu da internet of things (IoT) cihazları arasında veri aktarımını hızlandırır (Skaria et al. 2024). Ağın kötü niyetli düğüm barındırmasına karşı dayanıklılığı, sistemin güvenilirliğini artırır ve %33'e kadar (Commey et al. 2024). Deterministik yapısı, blok zincir ağındaki tutarsızlık risklerini azaltır ve işlemlerin belirli bir sırayla gerçekleşmesini sağlar (Tran et al. 2021). Private Transaction Manager (PTM) entegrasyonu, verilerin güvenliğini artırır çünkü yalnızca yetkilendirilmiş kullanıcılar işlemleri yapabilir (Martina et al. 2023).

5.5. Python tabanlı akıllı sözleşmelerin işleyişi

Hyperledger Besu platformu üzerinde çalışan Python tabanlı akıllı sözleşmeler, Web3.py kütüphanesi kullanılarak geliştirilmiştir. Web3.py, Ethereum tabanlı ağlarla Python üzerinden iletişim kurmayı sağlayan kapsamlı bir API'dir ve Hyperledger Besu'nun EVM (Ethereum Virtual Machine) uyumluluğu sayesinde sorunsuz şekilde entegre edilebilmektedir (Ray 2023) (Park et al. 2023) , (Martina et al. 2023), (Tran et

al. 2021). (Pradhan et al. 2022). Python tabanlı akıllı sözleşmelerin işlevleri aşağıdaki başlıklarda incelenebilir:

5.6. 128 KB’lik parçalama ve veri doğrulama süreci

Görüntü verilerinin blok zincir ağına eklenmesinde 128 KB'lik parçalama yöntemi kullanılmıştır. Parçalama yöntemi, özellikle büyük boyutlu verilerin aktarımı ve depolanmasında verimliliği artıran etkili bir yöntemdir. Bu yöntem, özellikle IoT sistemlerinde düşük bant genişliğine sahip ağlarda ve sınırlı işlem gücüne sahip cihazlarda veri aktarımının optimizasyonu için tercih edilmektedir (Durga et al. 2022; Mohanta et al. 2021), (Hasan et al. 2022; Xiong et al. 2020; Zhang et al. 2023).

5.7. Parçalama yönteminin işleyişi

128 KB'lik parçalama yöntemi, büyük boyutlu görüntü verisinin 128 KB’lık küçük parçalara bölünmesiyle başlamakta ve her bir parçanın bağımsız olarak işlenmesine olanak sağlayarak ağ trafiğinin daha verimli kullanılmasını sağlamaktadır. Parçalama işleminden sonra her bir parça için SHA-256 algoritması kullanılarak özetleme değeri oluşturulmakta ve bu özet değerleri blok zincir ağına eklenerek parçaların bütünlüğü garanti altına alınmaktadır (Wu et al. 2024; Yang et al. 2024; Zaidi et al. 2021). Veri kaydı aşamasında, her bir parçaya ait Chunk ID, dosya adı ve özetleme değeri gibi bilgiler blok zincir üzerinde depolanmakta ve doğrulama işlemlerinde bu bilgiler kullanılarak parçaların doğruluğu kontrol edilmektedir. Veri aktarımı sırasında herhangi bir parçanın eksik olduğu tespit edilirse, yalnızca eksik kalan parça yeniden gönderilmekte, böylece tüm verinin baştan iletilmesine gerek kalmadan ağ trafiği azaltılmakta ve işlem süresi önemli ölçüde iyileştirilmektedir.

5.7.1. Parçalama yönteminin avantajları

a) *Verimlilik:* Büyük boyutlu dosyaların küçük parçalara bölünmesi, ağ trafiğinin dengelenmesine ve dosya aktarımının daha hızlı tamamlanmasına katkı sağlar (Durga et al. 2022; Mohanta et al. 2021).
b) *Veri Bütünlüğü:* Her parçanın özetleme değeri hesaplanarak blok zincir ağına kaydedildiği için, olası manipülasyonlar hızlıca tespit edilebilir (Hasan et al. 2022; Xiong et al. 2020; Zhang et al. 2023).

c) *İşlem Süresi Optimizasyonu:* Eksik veya hatalı parçaların yalnızca ilgili parçanın tekrar gönderilmesi, dosyanın baştan sona yeniden aktarılmasını önleyerek zaman tasarrufu sağlar (Wu et al. 2024; Yang et al. 2024; Zaidi et al. 2021).
d) *Güvenlik:* Parçalanan veriler ayrı özetleme değerleriyle korunarak, yetkisiz müdahalelerin tespiti kolaylaştırılır (Hasan et al. 2022; Xiong et al. 2020; Zhang et al. 2023).

Yani bu yöntem, IoT sistemlerinde sıkça karşılaşılan bant genişliği kısıtlamalarını ve veri aktarım hatalarını minimize etmek amacıyla geliştirilmiştir. Özellikle akıllı şehir, trafik izleme sistemleri gibi geniş veri hacmi gerektiren uygulamalarda etkili bir çözüm sağlamaktadır (Durga et al. 2022; Mohanta et al. 2021), (Hasan et al. 2022; Xiong et al. 2020; Zhang et al. 2023).

3.4. Veri güvenliği ve erişim kontrolü

IoT cihazları tarafından alınan görüntülerin bütünlük ve doğruluk kontrolünü sağlamak amacıyla geliştirilmiştir. (Khan and Byun 2020), gizlilik ve güvenliği garanti altına almak için izinli bir blok zincir üzerinde endüstriyel IoT görüntü verilerini şifrelemek ve depolamak üzere bir şema önermektedir. Önerilen yapıda, görüntü verilerinin blok zincire kaydedilmesi, her görüntünün kriptografik özetleme fonksiyonu ile üretilen benzersiz kimliği olan özetleme değeri, görüntü dosyasının tanımlayıcı adı olan dosya adı ve görüntünün parçalı aktarımı sırasında her parçaya ait kimlik bilgisi olan chunk ID gibi bilgilerle birlikte gerçekleştirilmektedir. Bu verilerin blok zincir ağına kaydedilmesi, veri manipülasyonunun önüne geçmekte ve verinin herhangi bir şekilde değiştirilip değiştirilmediğini hızlı bir şekilde doğrulama olanağı sunmaktadır (Skaria et al. 2024).

5.7.2. Yetkilendirme ve erişim kontrolü

Blok zincir ağlarındaki akıllı sözleşmeler, çeşitli alanlarda güvenli ve merkezi olmayan veri erişim kontrolü için umut verici çözümler sunar (Li, Han, and Chang 2023). Akıllı sözleşme, yalnızca yetkili kullanıcıların verilere erişmesini sağlayacak şekilde tasarlanmıştır. Bu doğrultuda:

a) *Kullanıcı Yetkilendirme Mekanizması:* Blok zincir ağına erişim yetkisi olan kullanıcılar akıllı sözleşme üzerinden kaydedilmekte ve yalnızca bu kullanıcıların belirli verilere erişimine izin verilmektedir.

b) Erişim Kontrolü: Özel erişim kontrolleri ve token tabanlı kimlik doğrulama yöntemleri uygulanarak yetkisiz erişimler engellenmektedir (Commey et al. 2024).

Bu yapı, IoT sistemlerinde sıkça karşılaşılan veri sızıntılarını ve gizlilik ihlallerini minimize etmeyi hedeflemektedir.

5.7.3. Bütünlük kontrolü

Veri bütünlüğünün sağlanması için aşağıdaki doğrulama mekanizması uygulanmaktadır:

a) Özetleme Kontrolü: IoT cihazı tarafından gönderilen görüntünün özetleme değeri, blok zincire daha önce kaydedilen orijinal özetleme değeriyle karşılaştırılır.(Khor et al. 2023)

b) Veri Doğrulama: Eşleşme sağlanırsa verinin bütünlüğü doğrulanmış olur, aksi durumda manipülasyon veya hatalı aktarım tespit edilerek uyarı mekanizması devreye girer (Tran et al. 2021).

Bu doğrulama mekanizması sayesinde blok zincir ağı üzerinde saklanan verilerin değiştirilemezliği ve doğruluğu garanti altına alınmaktadır.

5.8. Test ortamı ve fiziksel kurulum

Bu çalışmada geliştirilen sistemin testleri, özel olarak kurulan bir mini şehir simülasyon alanında gerçekleştirilmiştir. TurtleBot3 ve Raspberry Pi cihazları, yapılandırılmış bir yol ağı üzerinde görüntü toplama ve blok zincire veri gönderme işlemlerini yerine getirmiştir. Test ortamında kullanılan yapay nesneler, sensörlerin algılama performansını test etmek ve gerçek dünya senaryolarını modellemek amacıyla yerleştirilmiştir. Aşağıda test ortamına ve kullanılan sistem bileşenlerine ait görseller sunulmaktadır:

Geliştirilen sistemin yerleştirildiği mini şehir yapısının genel görünümü Şekil 5.4'te sunulmuştur. Simülasyon ortamı, sensörlerin algılama doğruluğunu test edebilmek ve farklı senaryoları canlandırmak amacıyla çeşitli nesnelerle zenginleştirilmiştir.

Şekil 5. 4 Test nesneleri genel görünümü.

Şekil 5.5’ de geliştirilen sistemin gerçek dünyaya uygun şekilde LEGO tabanlı bir simülasyon alanında test edildiği ortam gösterilmektedir. TurtleBot3 robotu üzerine entegre edilen kamera ve sensör modülleri sayesinde çevresel görüntü verisi toplanmakta; bu veriler Raspberry Pi aracılığıyla özetleme’lenmekte ve HTTP RPC protokolü üzerinden Hyperledger Besu blok zincir ağına aktarılmaktadır. Görselde robot üzerindeki kamera ve sensör sistemleri, Raspberry Pi ünitesi ve kontrol amaçlı bağlı dizüstü bilgisayar görülmektedir.

Şekil 5. 5 TurtleBot3 robotu ile veri toplanan alanda kullanılan donanımlar.

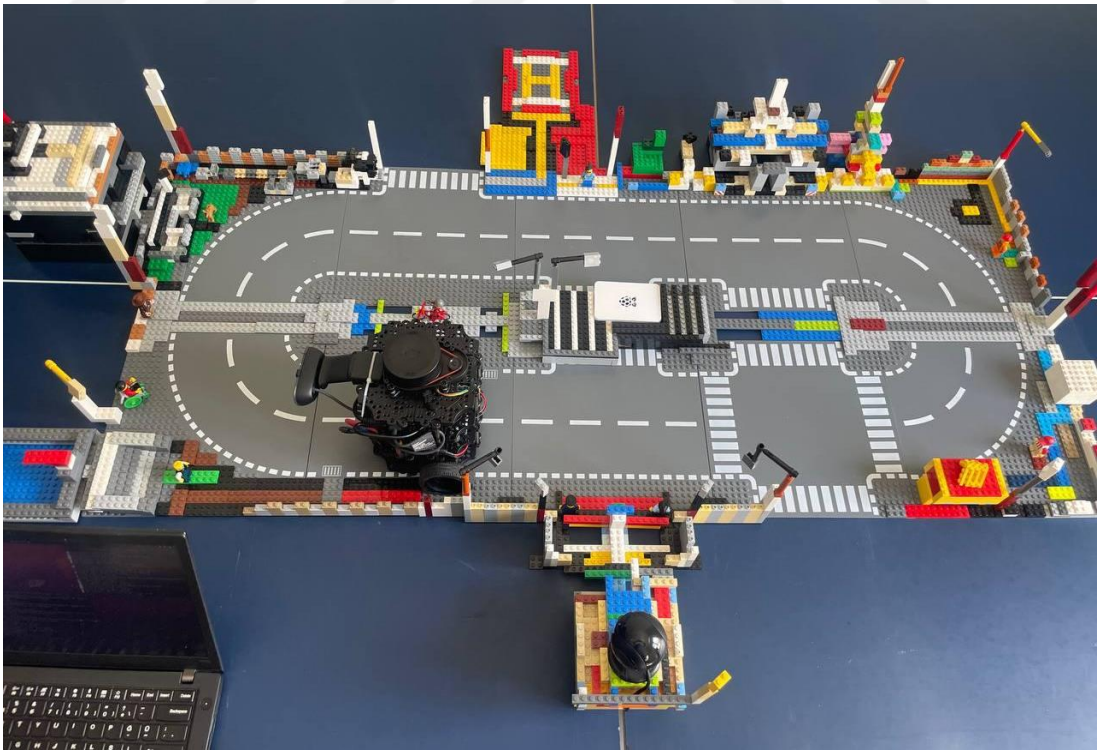
Şekil 5.6’ da ise LEGO ile modellenmiş yol, yaya geçidi, bina ve trafik öğeleri ile, TurtleBot3’ün gerçek hayattaki veri toplama ve aktarma senaryolarını test edebileceği bir ortam oluşturulmuştur. Böylece sistemin ölçeklenebilirliği, yol durumlarına adaptasyonu ve görev yönetimi değerlendirilebilmektedir. Sistem, bu verileri SHA-256 özetlemleriyle birlikte blok zincir ağına iletmektedir.

Şekil 5. 6 TurtleBot3, sürüş sırasında kaydettiği çevresel görüntü örneği.

Şekil 5.7, görüntü verilerinin parça-parça işlenme sürecine dair farklı açılardan elde edilen kamera görüntülerini göstermektedir. Her bir parça, blok zincire iletilmeden önce bütünlük doğrulamasıyla güvence altına alınmıştır.

Şekil 5. 7 IoT cihazından elde edilen görüntü.

Tüm sistemin fiziksel yerleşimini gösteren üstten bakış görüntüsü Şekil 5.8'de sunulmuştur. Yol ağı, çevresel objeler ve robotların konumları bu görselde detaylı biçimde yer almaktadır.



Şekil 5. 8 Test alanı üstten görünüm.

ALTINCI BÖLÜM

PERFORMANS DEĞERLENDİRMESİ VE ANALİZİ

Önerilen sistemin performans değerlendirmesi, parçalama yöntemi ve Hyperledger Besu'nun performansı gibi iki ana başlık altında ele alınmıştır. Yapılan analizlerde hem veri aktarım hızı hem de işlem süresi gibi kritik metrikler değerlendirilmiştir. Ayrıca önerilen sistemde TurtleBot ve Raspberry Pi cihazlarından alınan verilerin blok zincire eklenme süreci analiz edilmiştir. Yapılan testlerde TurtleBot'un ROS tabanlı hareket kontrolüyle eş zamanlı olarak görüntü aktarımının gecikme sürelerini artırmadığı gözlemlenmiştir. Raspberry Pi'den alınan verilerin ise düşük enerji tüketimi sayesinde kesintisiz şekilde blok zincire aktarıldığı tespit edilmiştir.

6.1. Parçalama yönteminin etkileri

Önerilen sistemde kullanılan 128 KB’lik parçalama yöntemi, büyük boyutlu görüntü dosyalarının blok zincir ağına aktarımı sırasında performans iyileştirmesi sağlamıştır. Parçalama yönteminin başlıca avantajları şunlardır:

6.1.1. Aktarım hızı artışı

Görüntü verilerinin 128 KB'lik parçalara ayrılması, blok zincir ağına yüklenme süresini azaltmıştır. Büyük boyutlu dosyaların tek parça halinde aktarılmasına kıyasla daha verimli sonuçlar elde edilmiştir. Özellikle IoT cihazlarının sınırlı bant genişliği göz önüne alındığında, bu yöntem daha hızlı ve kesintisiz veri aktarımını mümkün kılmıştır.

6.1.2. Veri kaybının önlenmesi

Parçalama sayesinde başarısız bir aktarımda yalnızca eksik parçanın yeniden gönderilmesi sağlanmıştır. Bu yaklaşım, ağ üzerindeki yükü azaltarak yeniden aktarım sürelerini optimize etmiştir. Ayrıca, bu yöntem IoT cihazlarının enerji tasarrufu yapmasını sağlamış ve sistemin dayanıklılığını artırmıştır.

6.2. Hyperledger Besu performans analizi

Hyperledger Besu'nun sağladığı IBFT 2.0 Mutabakat algoritması sayesinde aşağıdaki performans iyileştirmeleri sağlanmıştır:

Daha Düşük Gecikme (Latency), IBFT 2.0 Mutabakat algoritması, karar mekanizmasını hızlandırarak blok doğrulama sürelerini optimize etmiştir. Bu özellik, IoT cihazlarından gelen verilerin blok zincir ağına gecikmesiz kaydedilmesini sağlamıştır. Yapılan performans ölçümlerinde, blok üretim sürelerinin 0.5-2 saniye aralığında olduğu gözlemlenmiştir.

Veri Transferinde Yüksek Verimlilik, 4 düğümlük yapının kullanılması, yük dengeleme (load balancing) mekanizmasını iyileştirerek veri işleme sürecini hızlandırmıştır. Özellikle yüksek hacimli görüntü verilerinin transferinde, ağ üzerindeki trafiğin etkili şekilde yönetilmesi sayesinde gecikme ve veri kaybı önlenmiştir.

Tablo 6.1, Tablo 6.2 ve Tablo 6.3, sırasıyla Node-3 ve Node-4 üzerinde gerçekleştirilen veri işleme süreçlerine ait deneysel performans çıktılarının ayrıntılarını sunmaktadır. Her iki düğümde de görüntü parçalama, SHA-256 bütünlük doğrulama ve blok zincire veri gönderme işlemleri sırasında sistem kaynaklarının kullanımı detaylı olarak analiz edilmiştir.

6.3. İşlem süresi, bellek ve cpu analizi

IoT cihazlarından elde edilen 640x480 piksel çözünürlüğündeki görüntüler ile gerçekleştirilen performans ölçümleri, aşağıdaki tablolarda sunulmuştur:

Tablo 6. 1: Node3 için deneysel sonuçlar

Başlangıç Zamanı	Görüntü Yakalama Süresi (s)	Görüntü Boyutu (MB)	Bitiş Zamanı	İşlem Süresi (s)	İşlemci Kullanımı (%)	Bellek Kullanımı (%)	Disk Kullanımı (MB)
2025-03-19 17:18:56.355671 +03:00	0.71	0.1	2025-03-19 17:20:16.522772 +03:00	80.17	100.0% → 30.6%	57.6% → 59.2%	11621.97 MB → 11623.55 MB
2025-03-19 17:21:13.373599 +03:00	0.7	0.09	2025-03-19 17:21:29.890016 +03:00	16.52	100.0% → 41.5%	59.3% → 59.6%	11623.79 MB → 11623.96 MB
2025-03-19 17:21:57.053831 +03:00	0.7	0.08	2025-03-19 17:22:33.715404 +03:00	36.66	100.0% → 53.5%	59.6% → 60.3%	11624.66 MB → 11687.37 MB
2025-03-19 17:22:56.450633 +03:00	0.71	0.1	2025-03-19 17:23:55.389433 +03:00	58.94	100.0% → 41.4%	60.0% → 66.7%	11687.52 MB → 11689.25 MB
2025-03-19 17:24:16.743529 +03:00	0.71	0.1	2025-03-19 17:25:17.329260 +03:00	60.59	100.0% → 58.6%	67.1% → 68.5%	11689.48 MB → 11691.89 MB
2025-03-19 17:25:37.968959 +03:00	0.71	0.1	2025-03-19 17:25:58.750660 +03:00	20.78	100.0% → 38.6%	68.7% → 69.3%	11692.04 MB → 11692.31 MB

Tablo 6. 2: Node3 için deneysel sonuçlar (Devamı)

Başlangıç Zamanı	Görüntü Yakalama Süresi (s)	Görüntü Boyutu (MB)	Bitiş Zamanı	İşlem Süresi (s)	İşlemci Kullanımı (%)	Bellek Kullanımı (%)	Disk Kullanımı (MB)
2025-03-19 17:26:27.701750 +03:00	0.71	0.08	2025-03-19 17:26:58.980765 +03:00	31.28	100.0% → 57.8%	69.1% → 69.5%	11692.47 MB → 11693.61 MB
2025-03-19 17:27:20.326377 +03:00	0.71	0.09	2025-03-19 17:27:58.972174 +03:00	38.65	100.0% → 53.5%	69.5% → 69.6%	11693.71 MB → 11695.01 MB
2025-03-19 17:28:20.887345 +03:00	0.71	0.09	2025-03-19 17:29:03.012103 +03:00	42.12	100.0% → 47.0%	69.7% → 70.1%	11695.13 MB → 11696.32 MB
2025-03-19 17:29:28.694731 +03:00	0.71	0.11	2025-03-19 17:30:15.333398 +03:00	46.64	100.0% → 42.1%	70.0% → 70.4%	11696.48 MB → 11698.18 MB
2025-03-19 17:11:00.130721 +03:00	0.62	0.08	2025-03-19 17:11:16.213266 +03:00	16.08	100.0% → 48.8%	50.4% → 50.7%	12595.83 MB → 12596.05 MB
2025-03-19 17:11:35.441959 +03:00	0.63	0.08	2025-03-19 17:11:47.647731 +03:00	12.21	100.0% → 46.0%	50.6% → 51.0%	12596.25 MB → 12596.37 MB

Tablo 6. 3: Node4 için deneysel sonuçlar

Başlangıç Zamanı	Görüntü Yakalama Süresi (s)	Görüntü Boyutu (MB)	Bitiş Zamanı	İşlem Süresi (s)	İşlemci Kullanımı (%)	Bellek Kullanımı (%)	Disk Kullanımı (MB)
2025-03-19 17:12:09.654062 +03:00	0.64	0.07	2025-03-19 17:12:20.274053 +03:00	10.62	100.0% → 61.1%	51.0% → 51.2%	12596.57 MB → 12596.68 MB
2025-03-19 17:12:32.583070 +03:00	0.62	0.06	2025-03-19 17:12:51.712989 +03:00	19.13	100.0% → 55.6%	51.2% → 51.5%	12596.81 MB → 12597.96 MB
2025-03-19 17:13:11.684054 +03:00	0.62	0.09	2025-03-19 17:13:19.971856 +03:00	8.29	100.0% → 78.6%	51.5% → 51.8%	12598.23 MB → 12598.48 MB
2025-03-19 17:13:43.756355 +03:00	0.62	0.08	2025-03-19 17:14:20.876131 +03:00	37.12	100.0% → 50.9%	51.7% → 52.0%	12598.66 MB → 12599.99 MB
2025-03-19 17:14:50.007146 +03:00	0.62	0.08	2025-03-19 17:15:24.950349 +03:00	34.94	100.0% → 57.2%	52.0% → 52.3%	12600.21 MB → 12601.52 MB
2025-03-19 17:15:53.065280 +03:00	0.62	0.08	2025-03-19 17:16:06.683962 +03:00	13.62	100.0% → 45.4%	52.5% → 52.8%	12601.70 MB → 12601.82 MB

İlk olarak işlem süreleri incelendiğinde, Node-3 için ortalama işlem süresi ~36 saniye civarında seyrederken, Node-4' te bu sürenin daha düşük, ortalama ~25 saniye seviyelerinde olduğu gözlemlenmektedir. Bu fark, Node-4'ün daha optimize edilmiş veya daha az işlem yükü altında çalıştığına işaret etmektedir.

CPU kullanım oranları karşılaştırıldığında her iki düğümde de başlangıçta %100'e yakın CPU kullanımı gözlemlenmiş, işlem ilerledikçe bu oran azalmıştır. Bu durum, işlemlerin başta yoğun bir şekilde CPU gerektirdiğini, sonrasında ise daha çok veri aktarımı ve yazma işlemleri gibi I/O ağırlıklı adımlara kaydığını göstermektedir. Node-3' te CPU kullanımı %100’ den %30’lara kadar düşerken, Node-4' te bu düşüş daha belirgin şekilde %100’ den %45 seviyelerine inmektedir.

Bellek kullanım oranları açısından Node-3’ teki ortalama bellek kullanımı %52 ile %70 arasında değişmekte olup Node-4’ te bu oran %51 ila %56 arasında daha dar bir aralıkta seyretmektedir. Bu durum, Node-4' ün daha istikrarlı bir bellek yönetimi sağladığını göstermektedir.

HDD kullanımına bakıldığında ise Node-3' ün veri yazımı sırasında yaklaşık 0.6–0.8 MB aralığında HDD kullanımı gerçekleştirdiği, Node-4’ te ise bu değerin daha yüksek (yaklaşık 1.2–1.4 MB) olduğu görülmektedir. Bu fark, Node-4' ün daha büyük veya daha sık veri blokları oluşturduğunu, dolayısıyla zincire veri gönderme işlemini daha yoğun gerçekleştirdiğini düşündürmektedir.

Sonuç olarak; Node-4’ ün daha düşük işlem süresi, daha stabil bellek kullanımı ve daha yüksek disk yazım hacmi ile, blok zincir ağına veri aktarım süreçlerinde daha verimli bir profil sergilediği söylenebilir. Bu da mimari yapılandırma, yük dengeleme ya da donanım farklılıklarının sistem performansına etkisini ortaya koymaktadır. Dağıtık mimarilerde her bir düğümün performans analizinin yapılması, güvenilirlik ve süreklilik açısından kritik önem taşımaktadır.

6.4. Tartışmalı durumlar ve gözlemler

İşlemci kullanımı, blok zincire veri ekleme sürecinde CPU kullanımının %100'e yaklaştığı ancak işlem sonrasında belirgin şekilde düştüğü gözlemlenmiştir.

Bellek (RAM)kullanımı, görüntü parçalarının işlenmesi sırasında RAM kullanımının artış gösterdiği, ancak parçalama yöntemi sayesinde bu artışın stabil seviyede tutulduğu görülmüştür.

HDD alanı tüketimi, blok zincire eklenen her veri parçasının disk alanında belirli bir artışa neden olduğu, ancak parçalama yönteminin bu artışı optimize ettiği belirlenmiştir.

İşlem Süresi, IBFT 2.0 Mutabakat algoritması sayesinde blok doğrulama sürelerinin sabit kaldığı ve başarılı veri transfer oranının yüksek olduğu tespit edilmiştir.

Önerilen sistemin performans analizleri sonucunda, 128 KB'lik parçalama yöntemi ve IBFT 2.0 Mutabakat algoritması sayesinde sistemin verimli ve güvenli çalıştığı gözlemlenmiştir. Özellikle IoT cihazları gibi düşük donanım kapasitelerine sahip sistemlerde veri aktarımının kesintisiz ve güvenilir şekilde gerçekleştiği kanıtlanmıştır.

YEDİNCİ BÖLÜM

GÜVENLİK ANALİZİ VE SİBER TEHDİTLERE KARŞI DAYANIKLILIK

Sistem tasarımında güvenlik unsuru da bütüncül bir yaklaşımla ele alınmış ve potansiyel siber tehditlere karşı kapsamlı koruma sağlamak amacıyla çeşitli güvenlik mekanizmaları entegre edilmiştir. Bu kapsamda, önerilen sistem özellikle veri manipülasyonu, bağlantı kesintileri ve ağ tabanlı saldırılar gibi tehditlere karşı koruma sağlamaktadır. Python ile geliştirilen akıllı sözleşmelerde uygulanan *girdi doğrulama, hata yönetimi ve tekrar deneme mekanizması* gibi güvenlik önlemleri sayesinde sistem güvenliği artırılmıştır. *Girdi doğrulama* sürecinde, blok zincire kaydedilecek tüm veriler öncelikle doğrulama aşamasından geçirilmekte, özellikle görüntü özetleme değerleri, chunk ID ve dosya adı gibi kritik bilgiler üzerinde bütünlük kontrolü sağlanarak geçersiz veya manipüle edilmiş veri girişleri engellenmektedir. *Hata yönetimi* kapsamında ise, ağ kesintileri, doğrulama hataları ve yanıt gecikmeleri gibi olası hata durumları ele alınmakta ve sistemin kesintisiz çalışması sağlanmaktadır. Ayrıca, *tekrar deneme mekanizması* devreye alınarak bağlantı kopmaları veya ağ gecikmeleri nedeniyle gerçekleşemeyen işlemler yeniden gerçekleştirilmekte ve veri kaybı önlenerek işlem bütünlüğü korunmaktadır. Bu bütünleşik güvenlik yapısı sayesinde önerilen sistem, hem işlem güvenliği hem de veri bütünlüğü açısından yüksek güvenilirlik sunmaktadır. Bu kapsamda sistemin *DDoS saldırıları, veri manipülasyonu, yetkisiz erişim, veri kaybı ve kimlik avı* gibi tehditlere karşı aldığı önlemler Tablo 7.1'de özetlenmiştir.

Tablo 7. 1: Siber tehdit önlemleri.

Tehdit Türü	**Alınan Önlem**
DDoS Saldırıları	IBFT 2.0 Mutabakat algoritması ile yetkisiz node' ların spam blok üretimi engellenir.
Veri Manipülasyonu	Görüntü chunk' larının özetleme değerleri blok zincirde saklanarak veri bütünlüğü korunur.
Yetkisiz Erişim	Şifrelemesi ve roller/izin mekanizması ile yalnızca yetkili kullanıcıların erişimi sağlanır.
Veri Kaybı	Tekrar deneme mekanizması (Retry Mechanism) sayesinde bağlantı kopmalarında veri kaybı önlenir.
Kimlik Avı ve Sahte Cihazlar	Python tabanlı akıllı sözleşmeler, her cihaz için benzersiz kimlik doğrulama mekanizması içerir.

SEKİZİNCİ BÖLÜM

TARTIŞMA

Önerilen sistem, literatürde yer alan klasik metotlara kıyasla özellikle veri güvenliği, aktarım sürekliliği ve geliştirme esnekliği açısından önemli avantajlar sunmaktadır. Hyperledger Besu üzerinde uygulanan IBFT 2.0 mutabakat algoritması sayesinde, sistem manipülasyonlara karşı yüksek düzeyde koruma sağlamış ve veri güvenliği konusunda güçlü bir altyapı ortaya koymuştur. Görüntü verilerinin küçük parçalara ayrılarak gönderilmesini sağlayan parçalama yöntemi, aktarım sırasında yaşanabilecek kesintilerde yalnızca eksik parçanın yeniden gönderilmesine olanak tanımış, böylece hem veri kaybı minimize edilmiş hem de iletim hızı artırılmıştır. Geliştirme sürecinde Python programlama dili ile akıllı sözleşmelerin kullanılması ise, modüler mimari sayesinde sistemin geliştirilmesi kolaylaştırmış ve uygulama geliştirme sürecini hızlandırmıştır. Ancak, Python'un blok zinciri altyapılarında yaygın olarak tercih edilen Solidity veya Go gibi dillere kıyasla performans açısından bazı sınırlılıklar taşıdığı gözlemlenmiştir. Bu nedenle, sistemin gelecekte daha yüksek performansla çalışabilmesi için asenkron programlama teknikleri ve çoklu iş parçacığı gibi optimizasyonların entegre edilmesi planlanmaktadır. Elde edilen bu bulgular, önerilen mimarinin IoT tabanlı görüntü aktarımı senaryoları için etkili ve güvenilir bir çözüm sunduğunu ortaya koymakta ve ileride yapılacak geliştirmelere yönelik somut bir temel sağlamaktadır. Sistemin Hyperledger Besu altyapısı üzerine inşa edilmesi, güvenilirlik, veri bütünlüğü ve ölçeklenebilirlik açısından önemli katkılar sağlamıştır. IBFT 2.0 mutabakat algoritmasının sağladığı 2/3 çoğunluk onayı gereksinimi, ağ üzerinde gerçekleştirilen işlemlerde manipülasyon riskini azaltarak sistemin güvenliğini artırmıştır. Ayrıca, roller ve izinler yapısı sayesinde yalnızca yetkilendirilmiş kullanıcıların veri erişimine izin verilmiş ve bu sayede yetkisiz erişimlerin önüne geçilmiştir. Önerilen sistemde TurtleBot'un hareket halindeyken görüntü verisi toplama yeteneği, dinamik ortamlarda gerçek zamanlı veri ihtiyacını karşılayarak sistemin esnekliğini artırmıştır. Raspberry Pi'nin düşük enerji tüketimi sayesinde uzun süreli, sabit konumlu görüntü toplama işlemleri gerçekleştirilmiş ve bu iki cihazın birleşik yapısı sayesinde hem hareketli hem de sabit veri toplama senaryoları başarıyla yönetilmiştir. Hyperledger Besu'nun sağladığı hızlı ve güvenli işlem yapısı sayesinde, TurtleBot ve Raspberry Pi cihazlarından gelen veriler doğrudan blok zincir ağına entegre edilerek veri bütünlüğü korunmuştur. *Parçalama*

*yönteminin etkisi, b*üyük boyutlu görüntü dosyalarının blok zincir ağına aktarımı sırasında kullanılan 128 KB'lik parçalama yöntemi performans üzerinde doğrudan olumlu etkiler sağlamıştır. Bu yöntem, görüntülerin küçük parçalara bölünerek iletilmesini mümkün kılarak dosya boyutuna bağlı aktarım gecikmelerini azaltmıştır. IoT cihazlarının sınırlı bant genişliği ve işlem gücü göz önüne alındığında, parçalama yöntemi sistemin daha stabil ve verimli çalışmasını sağlamış, ayrıca başarısız aktarım durumlarında yalnızca eksik parçaların yeniden gönderilmesiyle ağ üzerindeki yük hafifletilmiştir. Böylece, tüm verinin baştan gönderilmesi gereksinimi ortadan kalkmış ve hem enerji hem de bant genişliği tasarrufu sağlanmıştır. Bu özellikler, özellikle düşük kaynaklı IoT cihazları için büyük avantajlar sunarak sistemin sürdürülebilirliğini artırmıştır. *Python programlama dili ile geliştirme yaklaşımı*, sistemin yazılım geliştirme süreçlerini hızlandırarak modüler yapı sayesinde bakım ve genişletme işlemlerini kolaylaştırmıştır. Web3.py kütüphanesinin sağladığı esneklik ve Python'un okunabilir yapısı, akıllı sözleşme geliştirme sürecini hem erişilebilir hem de yönetilebilir hale getirmiştir. Ancak Python'un Ethereum tabanlı platformlarda performans açısından optimize edilmemiş olması, özellikle büyük veri aktarımı ve işleme süreçlerinde bazı zorlukları beraberinde getirmiştir. Veri işleme sürecinde Python'un dinamik çalışma yapısı nedeniyle zaman zaman ek gecikmeler gözlemlenmiş, ayrıca büyük boyutlu görüntü verilerinin bellekte yönetilmesi sırasında kaynak kullanımında sınırlamalar ortaya çıkmıştır. Bu zorluklara rağmen, Python'un sunduğu geniş kütüphane desteği ve hızlı entegrasyon yetenekleri, sistemin geliştirilmesini kolaylaştırmış ve özellikle prototipleme süreçlerinde önemli bir avantaj sağlamıştır. Gelecekte performans iyileştirmeleri için Python'un asenkron programlama özelliklerinin ve çoklu iş parçacığı kullanımının entegrasyonu planlanmakta olup, böylece mevcut avantajlar korunurken performans sınırlamalarının da aşılması hedeflenmektedir. Tüm bu özellikler, önerilen sistemin hem akademik hem de endüstriyel uygulamalar için güçlü bir çözüm sunduğunu ortaya koymaktadır.

DOKUZUNCU BÖLÜM

SONUÇ VE GELECEK ÇALIŞMALAR

Bu çalışmada, blok zincir tabanlı IoT sistemlerinde görüntü verilerinin güvenli, mahremiyet odaklı ve verimli bir şekilde aktarılması için yenilikçi bir sistem önerilmiştir. Merkeziyetsiz blok zincir yapısı, Python tabanlı akıllı sözleşmeler ve gelişmiş kriptografik özetleme yöntemleri kullanılarak tasarlanan bu sistem, yüksek güvenlik gerektiren IoT uygulamaları için uygulanabilir bir çözüm sunmaktadır. Özellikle akıllı şehirler ve endüstriyel IoT gibi alanlarda, önerilen mimarinin güvenli veri aktarımı ve bütünlüğünü garanti etme kapasitesi yapılan analizlerle ortaya konmuştur. Gerçekleştirilen performans ve güvenlik değerlendirmelerinde, sistemin veri bütünlüğünü artırdığı, iletim sürekliliğini sağladığı ve geleneksel yöntemlere kıyasla olası saldırılara karşı daha dayanıklı olduğu tespit edilmiştir. Geliştirilen sistemin sunduğu bu güçlü özelliklere rağmen, dinamik gereksinimlere uyum sağlamak ve gelecekte ortaya çıkabilecek yeni tehditlere karşı daha dayanıklı hale getirmek amacıyla ek geliştirme ihtiyaçları da gündeme gelmiştir.

Gelecek çalışmalarda, önerilen sistemin esnekliğini artırmak ve daha geniş kullanım alanlarına uyarlamak için bazı kritik geliştirme başlıkları öne çıkmaktadır. Öncelikli olarak akıllı sözleşmelerin dinamik olarak güncellenebilmesini ve farklı IoT senaryolarına kolayca uyarlanabilmesini sağlamak üzere dinamik ve özelleştirilebilir akıllı sözleşme tasarımlarına yönelik çalışmalar yapılması planlanmaktadır (Liu et al. 2024). Ayrıca, kuantum bilgisayar tehditlerine karşı sistemin direncini artırmak için PQC algoritmalarının mutabakat süreçlerine ve veri işleme katmanlarına entegrasyonu araştırılacaktır (Commey et al. 2024). Görüntü verilerinin daha verimli saklanabilmesi ve blok zincir üzerindeki işlem yükünün azaltılması için gelişmiş veri sıkıştırma ve optimizasyon tekniklerinin entegrasyonu da performans iyileştirme hedefleri arasında yer almaktadır (Lu et al. 2020). Öte yandan, IoT veri akışlarının daha etkin analiz edilebilmesi ve güvenlik tehditlerinin erken aşamada tespit edilebilmesi için blok zincir tabanlı makine öğrenme algoritmalarının entegrasyonu da gelecek araştırmalar için önemli fırsatlar sunmaktadır (Skaria et al. 2024). Bu doğrultuda, IoT cihazlarının sınırlı işlem gücü ve enerji kapasitesi dikkate alınarak, hafif blok zincir protokollerinin geliştirilmesi de araştırma gündeminde yer almaktadır (Tran et al. 2021). Ayrıca, kullanıcı gizliliğini ve veri anonimliğini artırmak amacıyla homomorfik şifreleme, veri

maskeleme ve anonimleştirme gibi mahremiyet odaklı teknolojilerin sistemle bütünleştirilmesi, önerilen mimarinin kullanıcı güvenliğini daha da üst seviyeye taşıyacaktır (Commey et al. 2024).

Bu araştırma kapsamında önerilen blok zincir tabanlı IoT görüntü aktarım sistemi, veri bütünlüğünü, mahremiyeti ve güvenli paylaşımı sağlamak adına güçlü bir model ortaya koymuştur. Ancak, sistemin ölçeklenebilirliğinin artırılması ve farklı IoT ekosistemlerine uyarlanması için yapılacak ileri düzey araştırmalar, blok zincir ve IoT entegrasyonunun potansiyelini daha da genişletecektir. Enerji verimliliği, kuantum güvenliği ve yüksek hızlı veri işleme gibi ileri teknoloji başlıklarının detaylı olarak ele alınması, sistemin daha sürdürülebilir ve yüksek performanslı hale gelmesine katkı sağlayacaktır. Bu çalışma kapsamında geliştirilen çözüm, TurtleBot ve Raspberry Pi cihazlarından elde edilen görüntülerin güvenli ve verimli bir şekilde blok zincir ağına aktarılmasını sağlamış; böylece hareketli ve sabit IoT cihazları arasında güvenli veri akışının mümkün olduğunu göstermiştir. Gelecekte yapılacak çalışmalarda, TurtleBot'un hareket kontrol algoritmalarının daha da optimize edilerek veri iletim gecikmesinin azaltılması ve Raspberry Pi cihazlarının enerji tüketiminin minimuma indirilmesine yönelik iyileştirmeler planlanmaktadır. Bu geliştirmeler, önerilen sistemin hem performans hem de enerji verimliliği açısından daha da ileri seviyeye taşınmasını sağlayacak, gerçek dünya uygulamaları için daha uygun ve sürdürülebilir bir çözüm ortaya koyacaktır.

KAYNAKÇA

A. Fitwi and Y. Chen, "Secure and Privacy-Preserving Stored Surveillance Video Sharing atop Permissioned Blockchain," *2021 International Conference on Computer Communications and Networks (ICCCN)*, Athens, Greece, 2021, pp. 1-8, doi: 10.1109/ICCCN52240.2021.9522199.

Abbas, Hanaa, Maurantonio Caprolu, and Roberto Di Pietro. 2022. "Analysis of Polkadot: Architecture, Internals, and Contradictions." Pp. 61–70 in *2022 IEEE International Conference on Blockchain (Blockchain)*. IEEE.

Abbas, Hanaa, Maurantonio Caprolu, and Roberto Di Pietro. 2024. "Understanding Polkadot Through Graph Analysis: Transaction Model, Network Properties, and Insights." Pp. 259–75 in.

Abdella, Juhar, Zahir Tari, Adnan Anwar, Abdun Mahmood, and Fengling Han. 2021. "An Architecture and Performance Evaluation of Blockchain-Based Peer-to-Peer Energy Trading." *IEEE Transactions on Smart Grid* 12(4):3364–78. doi: 10.1109/TSG.2021.3056147.

Abdelmaboud, Abdelzahir, Abdelmuttlib Ibrahim Abdalla Ahmed, Mohammed Abaker, Taiseer Abdalla Elfadil Eisa, Hashim Albasheer, Sara Abdelwahab Ghorashi, and Faten Khalid Karim. 2022. "Blockchain for IoT Applications: Taxonomy, Platforms, Recent Advances, Challenges and Future Research Directions." *Electronics* 11(4):630. doi: 10.3390/electronics11040630.

Abd-El-Malek, Michael, Gregory R. Ganger, Garth R. Goodson, Michael K. Reiter, and Jay J. Wylie. 2005. "Fault-Scalable Byzantine Fault-Tolerant Services." Pp. 59–74 in *Proceedings of the twentieth ACM symposium on Operating systems principles*. New York, NY, USA: ACM.

Abdi, Adam Ibrahim, Fathy Elbouraey Eassa, Kamal Jambi, Khalid Almarhabi, and Abdullah Saad AL-Malaise AL-Ghamdi. 2020. "Blockchain Platforms and Access Control Classification for IoT Systems." *Symmetry* 12(10):1663. doi: 10.3390/sym12101663.

Al Ahmad, Mohammad A., Abdullah Al-Saleh, and Fahad A. Al Masoud. 2018. "Comparison between PoW and PoS Systems Of Cryptocurrency." *Indonesian Journal of Electrical Engineering and Computer Science* 10(3):1251. doi: 10.11591/ijeecs.v10.i3.pp1251-1256.

Alajlan, Razan, Norah Alhumam, and Mounir Frikha. 2023. "Cybersecurity for Blockchain-Based IoT Systems: A Review." *Applied Sciences* 13(13):7432. doi: 10.3390/app13137432.

Alam, Shadab. 2023a. "The Current State of Blockchain Consensus Mechanism: Issues and Future Works." *International Journal of Advanced Computer Science and Applications* 14(8). doi: 10.14569/IJACSA.2023.0140810.

Al-Jaroodi, Jameela, and Nader Mohamed. 2019. "Blockchain in Industries: A Survey." *IEEE Access* 7:36500–515. doi: 10.1109/ACCESS.2019.2903554.

Al-Shareeda, Mahmood A., Murtaja Ali Saare, and Selvakumar Manickam. 2023. "The Blockchain Internet of Things: Review, Opportunities, Challenges, and Recommendations." *Indonesian Journal of Electrical Engineering and Computer Science* 31(3):1673. doi: 10.11591/ijeecs.v31.i3.pp1673-1683.

Anon. 2003. *Proceedings of the General Track : 2003 USENIX Annual Technical Conference, June 9-14, 2003, San Antonio, Texas, USA*. USENIX Association.

Anon. 2018. *ICOIN 2018 : The 32nd International Conference on Information Networking : January 10-12, 2018, Holiday Inn ChiangMai, Chiang Mai, Thailand*. Institute of Electrical and Electronics Engineers.

Anon. 2019a. *2019 International Conference on Computer and Information Sciences (ICCIS) : Jouf University - Aljouf - Kingdom of Saudi Arabia, 03-04 April 2019*. IEEE.

Anon. 2019b. *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON) : Proceedings : Technology, Knowledge, and Society : 17-20 October 2019, Grand Hyatt Kochi Bolgatti, Kerala, India*. IEEE.

Anon. 2023. *Web3 for Better Whitepaper 3.0*.

Anon. n.d.-a. “Https://Besu.Hyperledger.Org/.”

Anon. n.d.-b. “Journal_jips_JIPS-2018-14-1-101 (1).”

Aponte, Fredy, Luz Gutierrez, Madga Pineda, Ines Meriño, Augusto Salazar, and Pedro Wightman. 2021. “Cluster-Based Classification of Blockchain Consensus Algorithms.” *IEEE Latin America Transactions* 19(4):688–96. doi: 10.1109/TLA.2021.9448552.

Aragon, Nicolas, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, and Gilles Zemor. 2022. “Ouroboros: An Efficient and Provably Secure KEM Family.” *IEEE Transactions on Information Theory* 68(9):6233–44. doi: 10.1109/TIT.2022.3168439.

ASANUMA, Takaki, and Takanori ISOBE. 2023. “MPoW: How to Make Proof of Work Meaningful.” *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* E106.A(3):2022CIP0010. doi: 10.1587/transfun.2022CIP0010.

Attaran, Mohsen, and Angappa Gunasekaran. 2019. *Applications of Blockchain Technology in Business*. Cham: Springer International Publishing.

Aublin, Pierre-Louis, Sonia Ben Mokhtar, and Vivien Quema. 2013. “RBFT: Redundant Byzantine Fault Tolerance.” Pp. 297–306 in *2013 IEEE 33rd International Conference on Distributed Computing Systems*. IEEE.

Bai, Chunguang, and Joseph Sarkis. 2020. “A Supply Chain Transparency and Sustainability Technology Appraisal Model for Blockchain Technology.” *International Journal of Production Research* 58(7):2142–62. doi: 10.1080/00207543.2019.1708989.

Baird, Leemon, Mance Harmon, and Paul Madsen. 2018. *Hedera: A Public Hashgraph Network & Governing Council*.

Baresi, Luciano, Giovanni Quattrocchi, Damian Andrew Tamburri, and Luca Terracciano. 2022. “A Declarative Modelling Framework for the Deployment and Management of Blockchain Applications.” Pp. 311–21 in

Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems. New York, NY, USA: ACM.

Bazzanella, Danilo, and Andrea Gangemi. 2023. "Bitcoin: A New Proof-of-Work System with Reduced Variance." *Financial Innovation* 9(1):91. doi: 10.1186/s40854-023-00505-2.

Bellaj, Badr, Aafaf Ouaddah, Emmanuel Bertin, Noel Crespi, and Abdellatif Mezrioui. 2022. "SOK: A Comprehensive Survey on Distributed Ledger Technologies." Pp. 1–16 in *2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. IEEE.

Bender, and Benedict. n.d. *PUBLIC BLOCKCHAIN-A SYSTEMATIC LITERATURE REVIEW ON THE SUSTAINABILITY OF CONSENSUS ALGORITHMS*.

Bhardwaj, Rashmi, and Debabrata Datta. 2020a. "Consensus Algorithm." Pp. 91–107 in.

Bosamia, Mansi, and Dharmendra Patel. 2020. "Comparisons of Blockchain Based Consensus Algorithms for Security Aspects." *International Journal on Emerging Technologies* 11(3):427–34.

Buchman, Ethan. 2016. *Tendermint: Byzantine Fault Tolerance in the Age of Blockchains*.

Burdges, Jeff, Alfonso Cevallos, Peter Czaban, Rob Habermeier, Syed Hosseini, Fabio Lama, Handan Kilinc Alper, Ximin Luo, Fatemeh Shirazi, Alistair Stewart, and Gavin Wood. 2020. "Overview of Polkadot and Its Design Considerations."

Buterin, Vitalik, Daniel Reijsbergen, Stefanos Leonardos, and Georgios Piliouras. 2019. "Incentives in Ethereum's Hybrid Casper Protocol." doi: 10.1002/nem.2098.

Buterin, Vitalik, Daniël Reijsbergen, Stefanos Leonardos, and Georgios Piliouras. 2020. "Incentives in Ethereum's Hybrid Casper Protocol." *International Journal of Network Management* 30(5). doi: 10.1002/nem.2098.

Cachin, Christian, and Marko Vukolić. 2017a. “Blockchain Consensus Protocols in the Wild.” in *Leibniz International Proceedings in Informatics, LIPIcs*. Vol. 91. Schloss Dagstuhl- Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing.

Capocasale, Vittorio, Danilo Gotta, and Guido Perboli. 2023. “Comparative Analysis of Permissioned Blockchain Frameworks for Industrial Applications.” *Blockchain: Research and Applications* 4(1):100113. doi: 10.1016/j.bcra.2022.100113.

Cäsar, Florian, Daniel P. Hughes, Josh Primero, and Stephen J. Thornton. n.d. *Cerberus A Parallelized BFT Consensus Protocol for Radix*.

Casino, Fran, Thomas K. Dasaklis, and Constantinos Patsakis. 2019. “A Systematic Literature Review of Blockchain-Based Applications: Current Status, Classification and Open Issues.” *Telematics and Informatics* 36:55–81. doi: 10.1016/j.tele.2018.11.006.

Chatziamanetoglou, Dimitrios, and Konstantinos Rantos. 2024. “Cyber Threat Intelligence on Blockchain: A Systematic Literature Review.” *Computers* 13(3).

Chen, Yu-Jia, Li-Chun Wang, and Shu Wang. 2020. “Stochastic Blockchain for IoT Data Integrity.” *IEEE Transactions on Network Science and Engineering* 7(1):373–84. doi: 10.1109/TNSE.2018.2887236.

Chenchev, Ivaylo. 2023a. “Classification of the DLT Consensus Algorithms with Focus on Blockchain.” Pp. 731–40 in.

Chitra, Tarun, Monica Quaintance, Stuart Haber, and Will Martino. 2019. “Agent-Based Simulations of Blockchain Protocols Illustrated via Kadena’s Chainweb.” Pp. 386–95 in *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE.

Clement, Allen, Edmund Wong, Lorenzo Alvisi, Mike Dahlin, and Mirco Marchetti. n.d. *Making Byzantine Fault Tolerant Systems Tolerate Byzantine Faults*.

Clincy, Victor, and Hossain Shahriar. 2019. “Blockchain Development Platform Comparison.” Pp. 922–23 in *Proceedings - International Computer Software and Applications Conference*. Vol. 1. IEEE Computer Society.

Commey, Daniel, Bin Mai, Sena G. Hounsinou, and Garth V. Crosby. 2024. “Securing Blockchain-Based IoT Systems: A Review.” *IEEE Access* 12:98856–81. doi: 10.1109/ACCESS.2024.3428490.

Çavdar Tuğrul, Ercüment Öztürk. 2018. “Nesnelerin İnterneti İçin Yeni Bir Mimari Tasarımı.” *Sakarya University Journal of Science* 22(1): 39–48. https://doi.org/10.16984/saufenbilder.285444.

Deepak, Preeti Gulia, Nasib Singh Gill, Mohammad Yahya, Punit Gupta, Prashant Kumar Shukla, and Piyush Kumar Shukla. 2024. “Exploring the Potential of Blockchain Technology in an IoT-Enabled Environment: A Review.” *IEEE Access* 12:31197–227. doi: 10.1109/ACCESS.2024.3366656.

Deng, Weichu, Teng Huang, and Haiyang Wang. 2022. “A Review of the Key Technology in a Blockchain Building Decentralized Trust Platform.” *Mathematics* 11(1):101. doi: 10.3390/math11010101.

Dib, Omar, Antoine Durand, Kei-Leo Brousmiche, Eric Thea, and Ben Hamida. 2018. *Consortium Blockchains: Overview, Applications and Challenges*.

Durga, R., E. Poovammal, Kadiyala Ramana, Rutvij H. Jhaveri, Saurabh Singh, and Byungun Yoon. 2022. “CES Blocks—A Novel Chaotic Encryption Schemes-Based Blockchain System for an IoT Environment.” *IEEE Access* 10:11354–71. doi: 10.1109/ACCESS.2022.3144681.

Dziembowski, Stefan, Sebastian Faust, Vladimir Kolmogorov, and Krzysztof Pietrzak. n.d. *Proofs of Space*.

Falazi, Ghareeb, Michael Hahn, Uwe Breitenbucher, Frank Leymann, and Vladimir Yussupov. 2019. “Process-Based Composition of Permissioned and Permissionless Blockchain Smart Contracts.” Pp. 77–87 in *Proceedings - 2019 IEEE 23rd International Enterprise Distributed Object Computing Conference, EDOC 2019*. Institute of Electrical and Electronics Engineers Inc.

Farshidi, Siamak, Slinger Jansen, Sergio Espana, and Jacco Verkleij. 2020. “Decision Support for Blockchain Platform Selection: Three Industry Case Studies.” *IEEE Transactions on Engineering Management* 67(4):1109–28. doi: 10.1109/TEM.2019.2956897.

Ferdous, Md Sadek, Mohammad Jabed Morshed Chowdhury, Mohammad A. Hoque, and Alan Colman. 2020. “Blockchain Consensus Algorithms: A Survey.”

Ferrag, Mohamed Amine, and Lei Shu. 2021. “The Performance Evaluation of Blockchain-Based Security and Privacy Systems for the Internet of Things: A Tutorial.” *IEEE Internet of Things Journal* 8(24):17236–60. doi: 10.1109/JIOT.2021.3078072.

De Filippi, Primavera, Morshed Mannan, and Wessel Reijers. 2020a. “Blockchain as a Confidence Machine: The Problem of Trust & Challenges of Governance.” *Technology in Society* 62:101284. doi: 10.1016/j.techsoc.2020.101284.

Fraga-Lamas, Paula, Tiago M. Fernández-Caramés, António M. Rosado Da Cruz, and Sergio Ivan Lopes. 2024. “An Overview of Blockchain for Industry 5.0: Towards Human-Centric, Sustainable and Resilient Applications.” *IEEE Access* 12:116162–201.

Gaba, Priyanka, Ram Shringar Raw, Mazin Abed Mohammed, Jan Nedoma, and Radek Martinek. 2022. “Impact of Block Data Components on the Performance of Blockchain-Based VANET Implemented on Hyperledger Fabric.” *IEEE Access* 10:71003–18. doi: 10.1109/ACCESS.2022.3188296.

Gervais, Arthur, Ghassan O. Karame, Vedran Capkun, and Srdjan Capkun. 2014. “Is Bitcoin a Decentralized Currency?” *IEEE Security and Privacy* 12(3):54–60. doi: 10.1109/MSP.2014.49.

Ghimire, Sarala, Jae Young Choi, and Bumshik Lee. 2020. “Using Blockchain for Improved Video Integrity Verification.” *IEEE Transactions on Multimedia* 22(1):108–21. doi: 10.1109/TMM.2019.2925961.

Guidi, Barbara, and Andrea Michienzi. 2023. “From NFT 1.0 to NFT 2.0: A Review of the Evolution of Non-Fungible Tokens.” *Future Internet* 15(6).

Guru, Abhishek, Bhabendu Kumar Mohanta, Hitesh Mohapatra, Fadi Al-Turjman, Chadi Altrjman, and Arvind Yadav. 2023. “A Survey on Consensus Protocols and Attacks on Blockchain Technology.” *Applied Sciences (Switzerland)* 13(4).

Hanis, Nurul, Abd Rasid, and O. K. Blockchain. n.d. *Blockchain Technology in E-Voting: Comparative Study*.

Haque, Ehtisham Ul, Waseem Abbasi, Ahmad Almogren, Jaeyoung Choi, Ayman Altameem, Ateeq Ur Rehman, and Habib Hamam. 2024. “Performance Enhancement in Blockchain Based IoT Data Sharing Using Lightweight Consensus Algorithm.” *Scientific Reports* 14(1):26561. doi: 10.1038/s41598-024-77706-x.

Hasan, Haya R., and Khaled Salah. 2018. “Proof of Delivery of Digital Assets Using Blockchain and Smart Contracts.” *IEEE Access* 6:65439–48. doi: 10.1109/ACCESS.2018.2876971.

Hasan, Haya R., Khaled Salah, Ibrar Yaqoob, Raja Jayaraman, Sasa Pesic, and Mohammed Omar. 2022. “Trustworthy IoT Data Streaming Using Blockchain and IPFS.” *IEEE Access* 10:17707–21. doi: 10.1109/ACCESS.2022.3149312.

Helliar, Christine V., Louise Crawford, Laura Rocca, Claudio Teodori, and Monica Veneziani. 2020. “Permissionless and Permissioned Blockchain Diffusion.” *International Journal of Information Management* 54. doi: 10.1016/j.ijinfomgt.2020.102136.

Hu, Bowen, Yingwen Chen, Hujie Yu, Linghang Meng, and Zhimin Duan. 2022. “Blockchain-Enabled Data-Sharing Scheme for Consumer IoT Applications.” *IEEE Consumer Electronics Magazine* 11(2):77–87. doi: 10.1109/MCE.2021.3066793.

Hu, Meng, Tao Shen, Jinbao Men, Zhuo Yu, and Yingli Liu. 2020. “CRSM: An Effective Blockchain Consensus Resource Slicing Model for Real-Time Distributed Energy Trading.” *IEEE Access* 8:206876–87. doi: 10.1109/ACCESS.2020.3037694.

Huh, Jun-Ho, and Seong-Kyu Kim. 2019. “The Blockchain Consensus Algorithm for Viable Management of New and Renewable Energies.” *Sustainability* 11(11):3184. doi: 10.3390/su11113184.

Hussein, Ziad, May A. Salama, and Sahar A. El-Rahman. 2023. “Evolution of Blockchain Consensus Algorithms: A Review on the Latest Milestones of Blockchain Consensus Algorithms.” *Cybersecurity* 6(1):30. doi: 10.1186/s42400-023-00163-y.

Imteaj, Ahmed, M. Hadi Amini, and Panos M. Pardalos. 2021. “Toward Smart Contract and Consensus Mechanisms of Blockchain.” Pp. 15–28 in.

Islam, Saminur, Mohammad Jaminur Islam, Mahmud Hossain, Shahid Noor, Kyung Sup Kwak, and S. M. Riazul Islam. 2023. “A Survey on Consensus Algorithms in Blockchain-Based Applications: Architecture, Taxonomy, and Operational Issues.” *IEEE Access* 11:39066–82. doi: 10.1109/ACCESS.2023.3267047.

Ismail, Leila, and Huned Materwala. 2019. “A Review of Blockchain Architecture and Consensus Protocols: Use Cases, Challenges, and Solutions.” *Symmetry* 11(10):1198. doi: 10.3390/sym11101198.

Jones, Karen L. 2019. *Blockchain: Building Consensus and Trust across the Space Sector*.

Juan Garay, Aggelos Kiayias, and Nikos Leonardos. 2015. *Advances in Cryptology - EUROCRYPT 2015*. Vol. 9057. edited by E. Oswald and M. Fischlin. Berlin, Heidelberg: Springer Berlin Heidelberg.

Khan, Prince Waqas, and Yungcheol Byun. 2020. “A Blockchain-Based Secure Image Encryption Scheme for the Industrial Internet of Things.” *Entropy* 22(2):175. doi: 10.3390/e22020175.

Khan, Shafaq Naheed, Faiza Loukil, Chirine Ghedira-Guegan, Elhadj Benkhelifa, and Anoud Bani-Hani. 2021. “Blockchain Smart Contracts: Applications, Challenges, and Future Trends.” *Peer-to-Peer Networking and Applications* 14(5):2901–25. doi: 10.1007/s12083-021-01127-0.

Khor, Jing Huey, Michail Sidorov, Ming Tze Ong, and Shen Yik Chua. 2023. "Public Blockchain-Based Data Integrity Verification for Low-Power IoT Devices." *IEEE Internet of Things Journal* 10(14):13056–64. doi: 10.1109/JIOT.2023.3259975.

Kim, Henry M., Hjalmar Turesson, Marek Laskowski, and Amir Fard Bahreini. 2022a. "Permissionless and Permissioned, Technology-Focused and Business Needs-Driven: Understanding the Hybrid Opportunity in Blockchain Through a Case Study of Insolar." *IEEE Transactions on Engineering Management* 69(3):776–91. doi: 10.1109/TEM.2020.3003565.

King, Sunny, and Scott Nadal. 2012. *PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake*.

Kotla, Ramakrishna, Lorenzo Alvisi, Mike Dahlin, Allen Clement, and Edmund Wong. 2009. "Zyzzyva." *ACM Transactions on Computer Systems* 27(4):1–39. doi: 10.1145/1658357.1658358.

Kshetri, Nir. 2017. "Blockchain's Roles in Strengthening Cybersecurity and Protecting Privacy." *Telecommunications Policy* 41(10):1027–38. doi: 10.1016/j.telpol.2017.09.003.

Lamport, Leslie. 2001. *Paxos Made Simple*.

Leng, Zeqi, Kunhao Wang, Yuefeng Zheng, Xiangyu Yin, and Tingting Ding. 2022. "Hyperledger for IoT: A Review of Reconstruction Diagrams Perspective." *Electronics* 11(14):2200. doi: 10.3390/electronics11142200.

Li, Dongcheng, W. Eric Wong, and Jincui Guo. 2020. "A Survey on Blockchain for Enterprise Using Hyperledger Fabric and Composer." Pp. 71–80 in *Proceedings - 2019 6th International Conference on Dependable Systems and Their Applications, DSA 2019*. Institute of Electrical and Electronics Engineers Inc.

Li, Hongzhi, Dezhi Han, and Chin-Chen Chang. 2023. "DAC4SH: A Novel Data Access Control Scheme for Smart Home Using Smart Contracts." *IEEE Sensors Journal* 23(6):6178–91. doi: 10.1109/JSEN.2023.3241093.

Li, Yunfa, Yifei Tu, Jiawa Lu, and Yunchao Wang. 2020. “A Security Transmission and Storage Solution about Sensing Image for Blockchain in the Internet of Things.” *Sensors* 20(3):916. doi: 10.3390/s20030916.

Lin, Iuon Chang, and Tzu Chun Liao. 2017. “A Survey of Blockchain Security Issues and Challenges.” *International Journal of Network Security* 19(5):653–59. doi: 10.6633/IJNS.201709.19(5).01.

Lin, Yangfei, Jie Li, Shigetomo Kimura, Yuanyuan Yang, Yusheng Ji, and Yangjie Cao. 2022. “Consortium Blockchain-Based Public Integrity Verification in Cloud Storage for IoT.” *IEEE Internet of Things Journal* 9(5):3978–87. doi: 10.1109/JIOT.2021.3102236.

Liu, Authors, Yizhi Liu, Xiaohan Hao, Wei Ren, Ruoting Xiong, Tianqing Zhu, Kim-Kwang Raymond Choo, Senior Member, and Geyong Min. 2022. *ORE Open Research Exeter TITLE A Blockchain-Based Decentralized, Fair and Authenticated Information Sharing Scheme in Zero Trust Internet-of-Things A NOTE ON VERSIONS A Blockchain-Based Decentralized, Fair and Authenticated Information Sharing Scheme in Zero Trust Internet-of-Things*.

Liu, Suhui, Jiguo Yu, Liquan Chen, and Baobao Chai. 2023. “Blockchain-Assisted Comprehensive Key Management in CP-ABE for Cloud-Stored Data.” *IEEE Transactions on Network and Service Management* 20(2):1745–58. doi: 10.1109/TNSM.2022.3185237.

Liu, Yang, Jinlong He, Xiangyang Li, Jingwen Chen, Xinlei Liu, Song Peng, Haohao Cao, and Yaoqi Wang. 2024. “An Overview of Blockchain Smart Contract Execution Mechanism.” *Journal of Industrial Information Integration* 41.

Lu, Yunlong, Xiaohong Huang, Yueyue Dai, Sabita Maharjan, and Yan Zhang. 2020. “Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial IoT.” *IEEE Transactions on Industrial Informatics* 16(6):4177–86. doi: 10.1109/TII.2019.2942190.

Mahajan, Deepa. 2019. “A Survey Paper on Blockchain Technology.” *International Journal for Research in Applied Science and Engineering Technology* 7(5):3564–69. doi: 10.22214/ijraset.2019.5584.

Makhdoom, Imran, Mehran Abolhasan, and Wei Ni. 2018. “Blockchain for IoT: The Challenges and a Way Forward.” Pp. 594–605 in. Scitepress.

Martina, Pa, R. Dhanvardini, R. Vijay, R. Amirtharajan, and Padmapriya Pravinkumar. 2023. “Design Development and Execution of Smart Contract : An Overview.” Pp. 1–5 in *2023 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE.

Mazi`eres, David, and Mazi` Mazi`eres. n.d. *The Stellar Consensus Protocol: A Federated Model for Internet-Level Consensus*.

Merrad, Yaçine, Mohamed Hadi Habaebi, Elfatih A. A. Elsheikh, Fakher Eldin M. Suliman, Md Rafiqul Islam, Teddy Surya Gunawan, and Mokhtaria Mesri. 2022. “Blockchain: Consensus Algorithm Key Performance Indicators, Trade-Offs, Current Trends, Common Drawbacks, and Novel Solution Proposals.” *Mathematics* 10(15).

Michelin, R. A., Ahmed, N., Kanhere, S. S., Seneviratne, A., & Jha, S. (2020). Leveraging lightweight blockchain to establish data integrity for surveillance cameras [arXiv preprint]. arXiv. https://doi.org/10.48550/arXiv.1912.11044

Mohanta, Bhabendu Kumar, Debasish Jena, Somula Ramasubbareddy, Mahmoud Daneshmand, and Amir H. Gandomi. 2021. “Addressing Security and Privacy Issues of IoT Using Blockchain Technology.” *IEEE Internet of Things Journal* 8(2):881–88. doi: 10.1109/JIOT.2020.3008906.

Monrat, Ahmed Afif, Olov Schelen, and Karl Andersson. 2019. “A Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities.” *IEEE Access* 7:117134–51. doi: 10.1109/ACCESS.2019.2936094.

Moolikagedara, K., Nguyen, M., Yan, W. Q., & Li, X. J. (2023). Video Blockchain: A Decentralized Approach for Secure and Sustainable Networks with Distributed Video Footage from Vehicle-Mounted Cameras

in Smart Cities. *Electronics*, *12*(17), 3621. https://doi.org/10.3390/electronics12173621

Moolikagedara, Kasun, Minh Nguyen, Weiqi Yan, and Xuejun Li. 2024. "Advancing Video Data Privacy Preservation in IoT Networks through Video Blockchain." *Information* 15(3):171. doi: 10.3390/info15030171.

Nakamoto, Satoshi. n.d. *Bitcoin: A Peer-to-Peer Electronic Cash System*.

Namane, Sarra, and Imed Ben Dhaou. 2022. "Blockchain-Based Access Control Techniques for IoT Applications." *Electronics* 11(14):2225. doi: 10.3390/electronics11142225.

Nguyen, Giang-Truong, and Kyungbaek Kim. 2018. "A Survey about Consensus Algorithms Used in Blockchain." *Journal of Information Processing Systems* 14(1):101–28. doi: 10.3745/JIPS.01.0024.

Nurhidayat, Wan, Wan Muhamad, Noorafiza Matrazali, Khairul Khalil Ishak, and Suzaimah Ramli. 2021. *Secure Blockchain-Based Data-Sharing Model and Adoption among Intelligence Communities*. Vol. 5.

Ogawa, Takeshi, Hayato Kima, and Noriharu Miyaho. 2018. "Proposal of Proof-of-Lucky-Id(PoL) to Solve the Problems of PoW and PoS." Pp. 1212–18 in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE.

Oh, Jintae, Joonyoung Park, Youngchang Kim, and Kiyoung Kim. 2020. "Algorithm Based on Byzantine Agreement among Decentralized Agents (BADA)." *ETRI Journal* 42(6):872–85. doi: 10.4218/etrij.2019-0489.

Ometov, Aleksandr, Yulia Bardinova, Alexandra Afanasyeva, Pavel Masek, Konstantin Zhidanov, Sergey Vanurin, Mikhail Sayfullin, Viktoriia Shubina, Mikhail Komarov, and Sergey Bezzateev. 2020. "An Overview on Blockchain for Smartphones: State-of-the-Art, Consensus, Implementation, Challenges and Future Trends." *IEEE Access* 8:103994–15. doi: 10.1109/ACCESS.2020.2998951.

Park, Andrew, Matthew Wilson, Karen Robson, Dionysios Demetis, and Jan Kietzmann. 2023. “Interoperability: Our Exciting and Terrifying Web3 Future.” *Business Horizons* 66(4):529–41. doi: 10.1016/j.bushor.2022.10.005.

Parssinen, Matti Antero, Mikko Kotila, Ruben Cuevas Rumin, Amit Phansalkar, and Jukka Manner. 2018. “Is Blockchain Ready to Revolutionize Online Advertising?” *IEEE Access* 6:54884–99. doi: 10.1109/ACCESS.2018.2872694.

Patel, Vishwani, Fenil Khatiwala, Kaushal Shah, and Yashi Choksi. 2020a. “A Review on Blockchain Technology: Components, Issues and Challenges.” Pp. 1257–62 in.

Pbc, Blockstack. 2020. *PoX: Proof of Transfer Mining with Bitcoin*.

Pradhan, Nihar Ranjan, Akhilendra Pratap Singh, Neeraj Kumar, Mohammad Mehedi Hassan, and Diptendu Sinha Roy. 2022. “A Flexible Permission Ascription (FPA)-Based Blockchain Framework for Peer-to-Peer Energy Trading With Performance Evaluation.” *IEEE Transactions on Industrial Informatics* 18(4):2465–75. doi: 10.1109/TII.2021.3096832.

Puthal, Deepak, Nisha Malik, Saraju P. Mohanty, Elias Kougianos, and Gautam Das. 2018. “Everything You Wanted to Know About the Blockchain: Its Promise, Components, Processes, and Problems.” *IEEE Consumer Electronics Magazine* 7(4):6–14. doi: 10.1109/MCE.2018.2816299.

Rahman, Anichur, Antonio Montieri, Dipanjali Kundu, Md Razaul Karim, Md Jahidul Islam, Sara Umme, Alfredo Nascita, and Antonio Pescapé. 2022. “On the Integration of Blockchain and SDN: Overview, Applications, and Future Perspectives.” *Journal of Network and Systems Management* 30(4). doi: 10.1007/s10922-022-09682-4.

Rathore, Heena, Amr Mohamed, and Mohsen Guizani. 2020. “A Survey of Blockchain Enabled Cyber-Physical Systems.” *Sensors* 20(1):282. doi: 10.3390/s20010282.

Ray, Partha Pratim. 2023. “Web3: A Comprehensive Review on Background, Technologies, Applications, Zero-Trust Architectures, Challenges and

Future Directions.” *Internet of Things and Cyber-Physical Systems* 3:213–48. doi: 10.1016/j.iotcps.2023.05.003.

Rico-Peña, Juan Jesús, Raquel Arguedas-Sanz, and Carmen López-Martin. 2023. “Models Used to Characterise Blockchain Features. A Systematic Literature Review and Bibliometric Analysis.” *Technovation* 123:102711. doi: 10.1016/j.technovation.2023.102711.

Rouhani, Sara, Rafael Belchior, Rui S. Cruz, and Ralph Deters. 2021. “Distributed Attribute-Based Access Control System Using Permissioned Blockchain.” *World Wide Web* 24(5):1617–44. doi: 10.1007/s11280-021-00874-7.

S., Velmurugan, Prakash M., Neelakandan S., and Eric Ofori Martinson. 2024. “An Efficient Secure Sharing of Electronic Health Records Using IoT-Based Hyperledger Blockchain.” *International Journal of Intelligent Systems* 2024:1–16. doi: 10.1155/2024/6995202.

Salimitari, Mehrdad, and Mainak Chatterjee. 2018. “A Survey on Consensus Protocols in Blockchain for IoT Networks.”

Sankar, Lakshmi Siva, M. Sindhu, and M. Sethumadhavan. 2017. “Survey of Consensus Protocols on Blockchain Applications.” Pp. 1–5 in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE.

Santhoshi, Sri, Devi Arigela, and Persis Voola. n.d. *Blockchain Open Source Tools: Ethereum and Hyperledger Fabric*.

Sayeed, Sarwar, and Hector Marco-Gisbert. 2019. “Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack.” *Applied Sciences* 9(9):1788. doi: 10.3390/app9091788.

Schwartz, David, Noah Youngs, and Arthur Britto. n.d. *The Ripple Protocol Consensus Algorithm*.

Segendorf, Björn. 2014. *What Is Bitcoin?*

Sharma, Jahanvi, Anju Sangwan, and Rishi Pal Singh. 2023. “A Review on Evolving Domains of <scp>Internet of Things</Scp> : Architecture,

Applications, and Technical Challenges." *International Journal of Communication Systems* 36(18). doi: 10.1002/dac.5613.

Sharma, Vishal, and Niranjan Lal. 2020. *A NOVEL COMPARISON OF CONSENSUS ALGORITHMS IN BLOCKCHAIN*. Vol. 20.

Shi, Long, Taotao Wang, Jun Li, Shengli Zhang, and Song Guo. 2023. "Pooling Is Not Favorable: Decentralize Mining Power of PoW Blockchain Using Age-of-Work." *IEEE Transactions on Cloud Computing* 11(3):2756–69. doi: 10.1109/TCC.2022.3226496.

Skaria, Teena, A. M. Anusha Bamini, and R. Chitra. 2024. "Secure Data Transmission in IIoT Using Blockchain Based BI-LSTM." Pp. 1–6 in *2024 International Conference on Advancement in Renewable Energy and Intelligent Systems (AREIS)*. IEEE.

Solat, Siamak, Philippe Calvez, and Farid Naït-Abdesselam. 2021. "Permissioned vs. Permissionless Blockchain: How and Why There Is Only One Right Choice." *Journal of Software* 95–106. doi: 10.17706/jsw.16.3.95-106.

Sun, Zhijie, Dezhi Han, Dun Li, Xiangsheng Wang, Chin-Chen Chang, and Zhongdai Wu. 2022. "A Blockchain-Based Secure Storage Scheme for Medical Information." *EURASIP Journal on Wireless Communications and Networking* 2022(1):40. doi: 10.1186/s13638-022-02122-6.

Szabo, Nick. 1997. "Formalizing and Securing Relationships on Public Networks." *First Monday* 2(9). doi: 10.5210/fm.v2i9.548.

Thakur, Supriya, and Vrushali Kulkarni. 2017. "Blockchain and Its Applications – A Detailed Survey." *International Journal of Computer Applications* 180(3):29–35. doi: 10.5120/ijca2017915994.

Touloupou, Marios, Marinos Themistocleous, Elias Iosif, and Klitos Christodoulou. 2022a. "A Systematic Literature Review Toward a Blockchain Benchmarking Framework." *IEEE Access* 10:70630–44. doi: 10.1109/ACCESS.2022.3188123.

Tran, Nguyen Khoi, M. Ali Babar, and Jonathan Boan. 2021. “Integrating Blockchain and Internet of Things Systems: A Systematic Review on Objectives and Designs.” *Journal of Network and Computer Applications* 173.

Verma, Neetu, Saurabh Jain, and Rajesh Doriya. 2021. “Review on Consensus Protocols for Blockchain.” Pp. 281–86 in *Proceedings - IEEE 2021 International Conference on Computing, Communication, and Intelligent Systems, ICCCIS 2021*. Institute of Electrical and Electronics Engineers Inc.

Verma, Sudhani, Divakar Yadav, and Girish Chandra. 2022a. “Introduction of Formal Methods in Blockchain Consensus Mechanism and Its Associated Protocols.” *IEEE Access* 10:66611–24. doi: 10.1109/ACCESS.2022.3184799.

Vivek Anand, M., and S. Vijayalakshmi. 2020. “Image Validation with Virtualization in Blockchain Based Internet of Things.” *Journal of Computational and Theoretical Nanoscience* 17(5):2388–95. doi: 10.1166/jctn.2020.8901.

Wang, Gerui, Jerome Wang, Liam Lai, and Fisher Yu. 2021. “Accountability and Forensics in Blockchains: XDC Consensus Engine DPoS 2.0.”

Wang, Wenbo, Dinh Thai Hoang, Peizhao Hu, Zehui Xiong, Dusit Niyato, Ping Wang, Yonggang Wen, and Dong In Kim. 2019. “A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks.” *IEEE Access* 7:22328–70. doi: 10.1109/ACCESS.2019.2896108.

Wu, Hao, Yu Liu, Konglin Zhu, and Lin Zhang. 2024. “Data-Sharing System with Attribute-Based Encryption in Blockchain and Privacy Computing.” *Symmetry* 16(11):1550. doi: 10.3390/sym16111550.

Wust, Karl, and Arthur Gervais. 2018. “Do You Need a Blockchain?” Pp. 45–54 in *Proceedings - 2018 Crypto Valley Conference on Blockchain Technology, CVCBT 2018*. Institute of Electrical and Electronics Engineers Inc.

XinFin (XDC) Hybrid Blockchain R&D Team. 2021. “XinFin Network-XDC Consensus Algorithm.”

Xing, Liudong. 2020. “Reliability in Internet of Things: Current Status and Future Perspectives.” *IEEE Internet of Things Journal* 7(8):6704–21. doi: 10.1109/JIOT.2020.2993216.

Xiong, Zehui, Yang Zhang, Nguyen Cong Luong, Dusit Niyato, Ping Wang, and Nadra Guizani. 2020. “The Best of Both Worlds: A General Architecture for Data Management in Blockchain-Enabled Internet-of-Things.” *IEEE Network* 34(1):166–73. doi: 10.1109/MNET.001.1900095.

Xu, Dingjie, Na Ren, and Changqing Zhu. 2023. “Integrity Authentication Based on Blockchain and Perceptual Hash for Remote-Sensing Imagery.” *Remote Sensing* 15(19):4860. doi: 10.3390/rs15194860.

Xu, Li Da, Yang Lu, and Ling Li. 2021. “Embedding Blockchain Technology Into IoT for Security: A Survey.” *IEEE Internet of Things Journal* 8(13):10452–73. doi: 10.1109/JIOT.2021.3060508.

Yadav, Ashok Kumar, and Karan Singh. 2020. “Comparative Analysis of Consensus Algorithms of Blockchain Technology.” Pp. 205–18 in *Advances in Intelligent Systems and Computing*. Vol. 1097. Springer.

Yang, Zenghui, Xiubo Chen, Yunfeng He, Luxi Liu, Yinmei Che, Xiao Wang, Ke Xiao, and Gang Xu. 2024. “An Attribute-Based Access Control Scheme Using Blockchain Technology for IoT Data Protection.” *High-Confidence Computing* 4(3):100199. doi: 10.1016/j.hcc.2024.100199.

Yao, Wei, Fadi P. Deek, Renita Murimi, and Guiling Wang. 2023a. “SoK: A Taxonomy for Critical Analysis of Consensus Mechanisms in Consortium Blockchain.” *IEEE Access* 11:79572–87. doi: 10.1109/ACCESS.2023.3298675.

Yuan, Yong, and Fei-Yue Wang. 2018. “Blockchain and Cryptocurrencies: Model, Techniques, and Applications.” *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 48(9):1421–28. doi: 10.1109/TSMC.2018.2854904.

Zaidi, Syed Yawar Abbas, Munam Ali Shah, Hasan Ali Khattak, Carsten Maple, Hafiz Tayyab Rauf, Ahmed M. El-Sherbeeny, and Mohammed A. El-Meligy. 2021. “An Attribute-Based Access Control for IoT Using Blockchain and Smart Contracts.” *Sustainability* 13(19):10556. doi: 10.3390/su131910556.

Zhang, Mingjin, Jiannong Cao, Yuvraj Sahni, Qianyi Chen, Shan Jiang, and Lei Yang. 2023. “Blockchain-Based Collaborative Edge Intelligence for Trustworthy and Real-Time Video Surveillance.” *IEEE Transactions on Industrial Informatics* 19(2):1623–33. doi: 10.1109/TII.2022.3203397.

Zhang, PeiYun, MengChu Zhou, QiXi Zhao, Abdullah Abusorrah, and Omaimah O. Bamasag. 2021. “A Performance-Optimized Consensus Mechanism for Consortium Blockchains Consisting of Trust-Varying Nodes.” *IEEE Transactions on Network Science and Engineering* 8(3):2147–59. doi: 10.1109/TNSE.2021.3079415.

Zhang, Shijie, and Jong-Hyouk Lee. 2020. “Analysis of the Main Consensus Protocols of Blockchain.” *ICT Express* 6(2):93–97. doi: 10.1016/j.icte.2019.08.001.

Zhang, Xiuxian, Xiaorong Zhu, and Inayat Ali. 2024. “Performance Analysis of IOTA Tangle and a New Consensus Algorithm for Smart Grids.” *IEEE Internet of Things Journal* 11(4):6396–6411. doi: 10.1109/JIOT.2023.3311103.

Zheng, Zibin, Shaoan Xie, Hong Ning Dai, Xiangping Chen, and Huaimin Wang. 2018. “Blockchain Challenges and Opportunities: A Survey.” *International Journal of Web and Grid Services* 14(4):352–75. doi: 10.1504/IJWGS.2018.095647.

Zhou, Sisi, Kuanching Li, Lijun Xiao, Jiahong Cai, Wei Liang, and Arcangelo Castiglione. 2023. “A Systematic Review of Consensus Mechanisms in Blockchain.” *Mathematics* 11(10):2248. doi: 10.3390/math11102248.

Zou, Yijun, Ting Meng, Peng Zhang, Wenzhen Zhang, and Huiyang Li. 2020. “Focus on Blockchain: A Comprehensive Survey on Academic and

Application.” *IEEE Access* 8:187182–201. doi: 10.1109/ACCESS.2020.3030491.

Zupan, Nejc, Kaiwen Zhang, and Hans-Arno Jacobsen. 2017. “Hyperpubsub: A Decentralized, Permissioned, Publish/Subscribe Service Using Blockchains.” Pp. 15–16 in *Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference: Posters and Demos*. New York, NY, USA: ACM.

ÖZGEÇMİŞ

Burak AĞGÜL, 2015 yılında İstanbul Sabahattin Zaim Üniversitesi Bilgisayar Mühendisliği Bölümünden lisans derecesini almıştır. 2021 yılında aynı üniversitenin Bilgisayar Bilimi ve Mühendisliği Anabilim Dalında yüksek lisans eğitimini tamamlamıştır. Doktora eğitimine ise yine İstanbul Sabahattin Zaim Üniversitesi Bilgisayar Bilimi ve Mühendisliği Anabilim Dalında devam etmektedir.

Akademik kariyerine Sakarya Uygulamalı Bilimler Üniversitesi'nde Öğretim Görevlisi olarak devam eden Burak AĞGÜL' ün çalışma alanları arasında blok zincir teknolojileri, görüntü işleme, derin öğrenme, yapay zeka ve mobil robotlar yer almaktadır.