

**BLOK ZİNCİRİ TEKNOLOJİSİNİN MATEMATİKSEL
TEMELLERİ VE BAZI UYGULAMALARI**

**MATHEMATICAL FUNDAMENTALS OF BLOCKCHAIN
TECHNOLOGY AND SOME APPLICATIONS**

ALİ İHSAN AKBULUT

DR. TALHA ARIKAN

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Matematik Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ

olarak hazırlanmıştır.

ÖZET

BLOK ZİNCİRİ TEKNOLOJİSİNİN MATEMATİKSEL TEMELLERİ VE BAZI UYGULAMALARI

Ali İhsan AKBULUT

Yükseklisans, Matematik Bölümü

Tez Danışmanı: Dr. Talha ARIKAN

Nisan 2025, 201 sayfa

Bu tezde, blok zinciri teknolojisi hem matematiksel temelleri hem de güncel uygulamalarıyla birlikte çok yönlü olarak ele alınmıştır. Çalışmada, blok zincirinin tarihçesi, teknik bileşenleri ve kriptografik altyapısı detaylı biçimde incelenmiş; özellikle özetleme fonksiyonları, dijital imzalar, Merkle ağaçları ve Eliptik Eğri Kriptografisi gibi yapılar üzerinden sistemin güvenliği ve veri bütünlüğü açıklanmıştır. Matematiksel yapıların sistem içindeki işlevi teknik düzeyde ve teorik güvenlik modelleri çerçevesinde değerlendirilmiştir.

Tezde ayrıca NFT, DeFi, DAO gibi blok zinciri uygulamalarının potansiyelleri tartışılmış, ancak kripto paralar dışındaki projelerin henüz geniş ölçekli yaygınlaşma sağlayamadığı vurgulanmıştır. Bu durumun, sistemin karmaşıklığı ve otorite dışı yapısının, yine mevcut otoritelerin desteğine ihtiyaç duyması gibi yapısal bir paradokstan kaynaklandığı değerlendirilmiştir. Özgün bir katkı olarak, blok zincire dayalı bir araç kiralama modeli önerilerek; tokenizasyon, akıllı sözleşmeler ve merkeziyetsiz yönetim ilkeleri doğrultusunda matematiksel temelli bir sistem tasarımı sunulmuştur.

Son bölümde, yapay zekâ ve Web 3.0 gibi yeni teknolojilerle blok zinciri arasındaki entegrasyon olanaklarına değinilmiş; özellikle yapay zekâ destekli çözümlerin kullanıcı deneyimini kolaylaştırarak sistemin yaygınlaşmasına katkı sağlayabileceğı belirtilmiştir. Bu tez, blok zincirini teknik, teorik ve uygulamalı yönleriyle ele alarak, gelecekteki disiplinler arası araştırmalara sağlam bir zemin oluşturulmasına katkı sağlamayı amaçlamaktadır.

Anahtar Kelimeler: Blok zinciri, Kriptografi, Hash Fonksiyonları, Dağıtık defter teknolojisi, Dijital imza, Tokenizasyon, Akıllı sözleşme, Merkeziyetsizlik, DAO, DeFi, NFT.



ABSTRACT

MATHEMATICAL FUNDAMENTALS OF BLOCKCHAIN TECHNOLOGY AND SOME APPLICATIONS

Ali İhsan AKBULUT

Master of Science, Mathematics

Supervisor: Dr. Talha ARIKAN

April 2025, 201 pages

In this thesis, blockchain technology is examined from a multifaceted perspective, encompassing both its mathematical foundations and contemporary applications. The study provides a detailed analysis of the history, technical components, and cryptographic infrastructure of blockchain systems, with a particular focus on hash functions, digital signatures, Merkle trees, and Elliptic Curve Cryptography as mechanisms that ensure system security and data integrity. The mathematical structures underlying these mechanisms are evaluated within a technical framework and theoretical security models.

The thesis also discusses the potential of blockchain applications such as NFTs, DeFi, and DAOs, emphasizing that projects beyond cryptocurrencies have not yet achieved widespread adoption. This limitation is attributed to a structural paradox: while blockchain is designed to operate independently of central authorities, its broader adoption still often requires support from those same authorities due to system complexity.

As an original contribution, the study proposes a blockchain-based car rental model, presenting a mathematically grounded system design based on tokenization, smart contracts, and principles of decentralized governance.

In the final section, the integration possibilities between blockchain and emerging technologies such as artificial intelligence and Web 3.0 are explored. It is argued that AI-powered solutions can enhance user experience and contribute to the broader adoption of blockchain systems. By addressing blockchain from technical, theoretical, and applied perspectives, this thesis aims to provide a solid foundation for future interdisciplinary research.

Keywords: Blockchain, Cryptography, Hash Functions, DLT, Digital Signature, Tokenization, Smart Contract, Decentralization, DAO, DeFi, NFT .



TEŞEKKÜR

Lisansüstü eğitimim süresince bilgi birikimi, akademik rehberliği ve kıymetli desteğiyle her zaman yanımda olduğunu hissettiren, yol gösterici tutumuyla bana ilham veren değerli danışman hocam Dr. Talha ARIKAN'a

Hayatım boyunca her koşulda yanımda olan, desteklerini ve sevgilerini hiçbir zaman esirgemeyen, fedakârlıkları ve anlayışlarıyla bana güç veren, bu süreçte en büyük motivasyon kaynağım olan canım aileme,

Özellikle de bu tezin tamamlanması sürecinde oyun saatlerinden çalan bu satırları hoş gören minik meleğim Zeynep Duru'ya,

Teşekkür ederim.

Ali İhsan AKBULUT

Mayıs 2025, İstanbul

İÇİNDEKİLER

ÖZET	i
ABSTRACT	iii
TEŞEKKÜR	v
İÇİNDEKİLER	vi
ŞEKİLLER DİZİNİ	ix
ÇİZELGELER DİZİNİ	x
KODLAR DİZİNİ	xi
KISALTMALAR	xii
1. GİRİŞ	1
2. BLOK ZİNCİRİ TEKNOLOJİSİ	5
2.1. Blok Zinciri Teknolojisinin Tanımı	5
2.2. Blok Zincirinin Tarihçesi ve Gelişimi	7
2.2.1. Blok Zinciri Hakkında Dikkat Çekici Olaylar ve Kritik Vakalar	13
2.3. Blok Zinciri Teknolojisinin Felsefesi ve Temel Kavramları	18
2.3.1. Merkeziyetsizlik (Decentralization)	20
2.3.2. Dağıtık Defter Teknolojisi (DLT)	22
2.3.3. Kriptografi	25
2.3.4. Şeffaflık ve İzlenebilirlik	28
2.3.5. Değiştirilemezlik (Immutability)	29
2.3.6. Konsensüs Mekanizmaları	30
2.3.7. Cüzdanlar (Wallet)	32
2.3.8. Borsalar	35
2.3.9. Akıllı Sözleşmeler	38
2.3.10. Katmanlar	39
2.3.11. Ölçeklenebilme ve trilemma	39
3. BLOK ZİNCİRİ TEKNOLOJİSİNE MATEMATİKSEL BİR BAKIŞ	43

3.1. Özetleme Fonksiyonları.....	43
3.1.1. Özetleme fonksiyonlarının veri bütünlüğü sağlama rolü	45
3.1.2. SHA-256 ve bazı diğer özetleme fonksiyonları	47
3.1.3. Özetleme fonksiyonlarının güvenliği	52
3.2. Simetrik Kriptografi	53
3.3. Asimetrik Kriptografi	55
3.3.1. Asimetrik Kriptografinin Tanımı.....	55
3.3.2. Şifreleme ve Şifre Çözme.....	56
3.3.3. Dijital İmzalar.....	57
3.3.4. Asimetrik Kriptografide Anahtarlar	58
3.3.5. Bazı Önemli Asimetrik Kriptografi Algoritmaları	59
3.3.6. Asimetrik Kriptografinin Geleceği.....	77
3.4. Protokoller	78
3.4.1. Taahhüt Protokolü	78
3.4.2. Kimlik Doğrulama Protokolleri.....	79
3.4.3. Sıfır Bilgi İspatları.....	79
3.4.4. Eşik (Threshold) Kriptografi	81
3.4.5. Güvenli Çok Taraflı Hesaplama(Secure Multi-Party Computation, SMPC)	81
3.5. Blok Zincirinde Kriptografi.....	82
3.5.1. Blok Zinciri Bileşenleri	83
3.5.2. Dijital İmza Algoritmaları ve Blok zinciri	85
3.5.3. Fikir Birliği (Konsensüs) Algoritmaları	88
3.5.4. Blok Zinciri Teknolojisinin Bazı Sıkıntıları.....	94
4. BLOK ZİNCİRİ TEKNOLOJİSİNİN ÇALIŞMA MODELİ.....	96
4.1. Modelin Bitcoin Özelinde Ayrıntılı İncelenmesi	96
4.2. Ethereum ve Akıllı Kontratlar	120
4.3. Geleneksel İşlemlerle Blok Zinciri Teknolojisinin Karşılaştırılması.....	137
5. BLOK ZİNCİRİ TEKNOLOJİSİNİN UYGULAMALARI.....	141
5.1. Kripto Paralar	142
5.2. Token ve NFT.....	154
5.3. Köprü Yapıları (Bridges).....	155
5.4. Merkeziyetsiz Finans (DeFi) ve Uygulamaları	156

5.5. Merkeziyetsiz Otonom Organizasyonlar (DAO'lar)	158
5.6. Sağlık Hizmetleri	158
5.7. Tedarik Zinciri Yönetimi	159
5.8. Kamu Hizmetleri ve Yönetim	160
6. BLOK ZİNCİRİ VE KRİPTO PARALARIN GELECEĞİ	162
6.1. ETF (Exchange Traded Fund – Borsa Yatırım Fonu)	162
6.2. Metaverse	163
6.3. Web 3.0	164
6.4. IoT (Internet of Things - Nesnelerin İnterneti)	165
6.5. Yapay Zekâ	166
6.6. Yasal Düzenlemeler	168
7. TOKENİZE ÖNERİSİ	172
7.1. Sektör seçimi	172
7.2. Platform seçimi	172
7.3. Araçların tokenizasyonu	173
7.4. Merkeziyetsiz Organizasyon Yapısı (DAO)	175
7.5. Akıllı Sözleşmeler ve İş Akışları	177
8. SONUÇ VE DEĞERLENDİRMELER	189
KAYNAKLAR	194
ÖZGEÇMİŞ	201

ŞEKİLLER DİZİNİ

Şekil 1: Bitcoin Fiyat (USD) Zaman Grafiği	10
Şekil 2: Blok zinciri teknolojilerinde ölçeklenebilme soru	40
Şekil 3: Blok zinciri trilemması.....	42
Şekil 4: Özetleme fonksiyonları ile mesajın değişmediğinin kontrolü.....	46
Şekil 5: Merkle-Damgard yapısı	47
Şekil 6 : Sha-256 Algoritmasında f fonksiyonunun bir raunt süreci	50
Şekil 7: Simetrik şifreleme	54
Şekil 8: Asimetrik şifreleme	56
Şekil 9: Dijital imza çalışma mekanizması	58
Şekil 10: Diffie-Hellman Anahtar Anlaşma Protokolü	60
Şekil 11: RSA şifreleme mekanizması	62
Şekil 12: $y^2 = x^3 - 4x + 2$ ve $y^2 = x^3 + 2x + 1$ düzgün eliptik eğrilerinin grafiği	64
Şekil 13: $y^2 = x^3 - 3x + 2$ tekil eliptik eğrinin grafiği	65
Şekil 14: Eliptik eğri üzerinde dikey olmayan iki farklı noktanın toplamı	66
Şekil 15: Eliptik eğri üzerinde bir noktanın kendisiyle toplanması	67
Şekil 16: $p = 19$ ve $y^2 = x^3 + 3x + 5$ eğrisinin $\mathbb{Z}_p \times \mathbb{Z}_p$ düzlemindeki grafiği	70
Şekil 17: Eliptik eğri kriptografisi ile bir mesajın şifrlenmesi.....	72
Şekil 18 : Dijital imza algoritması (DSA)	74
Şekil 19: Eliptik eğri tabanlı dijital imzalama algoritması (ECDSA)	76
Şekil 20: Blok hash değerlerinin bir sonraki blokta yer alması.....	83
Şekil 21: Bitcoin zincirinin 889,503. bloğu.....	84
Şekil 22: Merkle ağaç yapısı	90
Şekil 23 : Tam bir bitcoin düğümü.....	98
Şekil 24 : Genişletilmiş bitcoin ağı.	99
Şekil 25: Bitcoin de blok yapısı.....	102
Şekil 26: IBM FoodTrust Projesi	160
Şekil 27: Araçların tokenleştirilmesi	174
Şekil 28: Merkeziyetsiz Organizasyon karar süreci	176
Şekil 29: Araç satın alma süreci	179
Şekil 30: Araç kiralama süreci.....	181
Şekil 31: Araç satış süreci	183

ÇİZELGELER DİZİNİ

Tablo 1: Blok zinciri önemli siber saldırı olayları.....	18
Tablo 2: Bazı blok zinciri teknolojilerinin merkeziyetsizlik düzeyleri	22
Tablo 3 : Blok zincirinde farklı dağıtık defter yapıları.....	25
Tablo 4: Günlük işlem hacmine göre kripto borsaları	37
Tablo 5: Bazı Bitcoin Core sürümleri ve uyguladığı BIP'ler	96
Tablo 6: Bitcoin adresleri versiyonları.	109
Tablo 7: Bitcoin halving tablosu	114
Tablo 8: Ethereum hard fork'ları.....	122
Tablo 9: Ethereum istemcileri	124
Tablo 10: Ethereum Yellow Paper Appendix G'den işlem ve gas maliyetleri listesi.....	131
Tablo 11: Solidity ile yazılmış bir akıllı sözleşmenin incelenmesi	133
Tablo 12: Araçların tokenleştirme süreci kodlama yol haritası.....	175
Tablo 13: Merkeziyetsiz organizasyon yapısı kodlama yol haritası	176
Tablo 14: Araç satın alma süreci kodlama yol haritası	179
Tablo 15: Araç kiralama süreci kodlama yol haritası.....	182
Tablo 16: Araç satış süreci kodlama yol haritası.....	184
Tablo 17: Akıllı sözleşme fonksiyonları/işlemleri için tahmini gas tüketimi	185
Tablo 18: fonlamaBaslat fonksiyonu gas tüketimi hesaplaması	187
Tablo 19: fonlamaKatilim fonksiyonu gas tüketimi hesaplaması	187

KODLAR DİZİNİ

Kod 1: Bitcoin blok başlığı parametreleri	100
Kod 2: Merkle Kökü değerini hesaplayan kod parçası.....	101
Kod 3: Merkle algoritmasının nasıl çalıştığını gösteren kod parçası	102
Kod 4: Segwit güncellemesi ile ilgili tanımlanan yeni değişkenler	104
Kod 5: Segwit güncellemesi ile blok ağırlık hesaplama yöntemi	104
Kod 6: Blok için süre değişkenleri	105
Kod 7: Blok için zorluk hesaplaması.....	105
Kod 8: Bitcoin’de Coin değişkeninin yapısı	108
Kod 9: Bitcoin’de UTXO yapısı.....	108
Kod 10: Bir Bitcoin 100.000.000 Satoshi’dır.....	113
Kod 11: Halving süreci ile ilgili kod parçaları	113
Kod 12: Düzenlenebilir işlem parametreleri	116
Kod 13: İşlemin imzalanacak olan hash’ini oluşturan fonksiyon	117
Kod 14: İmzalama işlemini gerçekleştiren fonksiyon	117
Kod 15: Bir işlemin sabitlenmiş işlem parametreleri	118
Kod 16: Bir işlemin havuza alınması ve yapılan kontroller	118
Kod 17: Madencinin blok oluşturarak işlemleri kendi bloğuna seçmesi.....	119
Kod 18: İşlemlerin madenci bloğundan blok zinciri bloğuna yazılması	119
Kod 19: Ethereum blok başlıklarını gösteren kod parçası.....	125
Kod 20: Ethereum’da bir Legacy (Eski Tip) işlemin parametreleri.....	126
Kod 21: Ethereum’da bir EIP-1559 işlemin parametreleri.....	127
Kod 22: Ethereum’da bir EIP-2930 AccessListTx işlemin parametreleri.....	127

KISALTMALAR

API	Application Programming Interface
ASIC	Application-Specific Integrated Circuit
AVAX	Avalanche Kripto Para Birimi
BTC	Bitcoin
DAO	Decentralized Autonomous Organization (Merkeziyetsiz Otonom Organizasyon)
DAG	Directed Acyclic Graph (Yönlendirilmiş Asiklik Grafik)
DEX	Decentralized Exchange (Merkeziyetsiz Borsa)
DLT	Distributed Ledger Technology (Dağıtık Defter Teknolojisi)
ECC	Elliptic Curve Cryptography (Eliptik Eğri Kriptografisi)
ECDSA	Elliptic Curve Digital Signature Algorithm
EIP	Ethereum Improvement Proposal (Ethereum İyileştirme Önerisi)
EOA	Externally Owned Account (Dışarıdan Sahipli Hesap)
ERC	Ethereum Request for Comment (Ethereum Standartları)
ETF	Exchange-Traded Fund (Borsa Yatırım Fonu)
ETH	Ethereum
EVM	Ethereum Virtual Machine
ICO	Initial Coin Offering (İlk Coin Arzı)
IoT	Internet of Things (Nesnelerin İnterneti)
IPFS	InterPlanetary File System
NFT	Non-Fungible Token (Eşsiz Dijital Varlık)
PBFT	Practical Byzantine Fault Tolerance
RPC	Remote Procedure Call
SHA	Secure Hash Algorithm
SNARK	Succinct Non-Interactive Argument of Knowledge
SPV	Simplified Payment Verification
UTXO	Unspent Transaction Output
USDT	Tether (USD'ye sabitlenmiş stabil kripto para)
USDC	USD Coin (Stabil kripto para)
VR	Virtual Reality (Sanal Gerçeklik)
Web 3.0	Üçüncü Nesil İnternet
ZKP	Zero-Knowledge Proof (Sıfır Bilgi İspatı)

1. GİRİŞ

İnsanoğlunun tarih boyunca sürdürdüğü ilerleme çabası, her dönemde yeni keşiflerin yapılmasına ve bu keşiflerin üzerine inşa edilen gelişmişlik düzeylerinin oluşmasına zemin hazırlamaktadır. Kanaatimizce, içinde bulunduğumuz çağ, yalnızca mekanik ve teknolojik ilerlemelerle sınırlı kalmamakta; aynı zamanda yapay zekâ, blok zinciri, nesnelerin interneti (IoT) gibi dönüşüm yaratan teknolojilerle bireylerin, kurumların ve toplumların algı ve etkileşim biçimlerini kökten değiştirmektedir. Özellikle insan benzeri düşünme yeteneğine sahip yapay zekâ sistemleri, birçok alanda iş süreçlerini yeniden tanımlarken; blok zinciri teknolojisi, veri güvenliği, mülkiyet, doğrulama ve merkeziyetsizlik gibi konularda yeni paradigmaların ortaya çıkmasına neden olmaktadır. Bu süreçlerin sağlıklı işlemesi ise, üretilen devasa verinin güvenli biçimde saklanması, değiştirilemez olması, doğrulanabilmesi ve merkezi otoritelerden bağımsız olarak yönetilebilmesi gibi özellikleri zorunlu kılmaktadır. Bu gereklilikleri karşılamada günümüzde en güçlü aday olarak karşımıza çıkan blok zinciri teknolojisi, yalnızca teknik değil, aynı zamanda sosyal ve ekonomik etkileriyle de mevcut yapıları ve alışkanlıkları zorlayan bir paradigma değişimini beraberinde getirmektedir.

Blok zinciri teknolojisi, iş yapma biçimlerinden kurumsal organizasyonlara kadar pek çok alanda dönüştürücü etkiler yaratmakta; bireylerin, kurumların ve devletlerin rollerini yeniden kurgulamayı zorlamaktadır. Bununla birlikte, yapılan gözlemler blok zinciri kavramının, özellikle popülerleşen Bitcoin gibi kripto paralar aracılığıyla geniş kesimlerce duyulmuş olmasına rağmen, sistemin temel ilkeleri, matematiksel altyapısı ve gerçek potansiyeline dair bilginin oldukça sınırlı olduğunu göstermektedir. Mevcut kaynakların çoğunlukla popüler, spekülatif ya da yüzeysel düzeyde bilgi sunması, bu teknolojinin bütüncül bir bakış açısıyla anlaşılmasını zorlaştırmaktadır. Bu bağlamda, blok zinciri teknolojisinin teknik altyapısı, matematiksel temelleri ve sosyo-ekonomik boyutlarını da içeren çok yönlü bir derleme tezin hazırlanmasının gerekli olduğu değerlendirilmiş; bu amaçla söz konusu çalışma gerçekleştirilmiştir.

Tezde, konunun çok kapsamlı yapısı göz önünde bulundurularak temel olarak şu sorulara yanıt aranmaktadır: Blok zinciri teknolojisi nedir? Neden ihtiyaç duyulmuştur? Matematiksel altyapısı nasıl çalışır? Ne tür uygulama alanları vardır? Mevcut sistemlerle nasıl etkileşim hâlinindedir? Sağladığı fırsatlar ve barındırdığı riskler nelerdir? Bu çerçevede blok zinciri sisteminin yalnızca teknik yönleri değil, aynı zamanda etik, psikolojik ve toplumsal etkilerine de değinilmiştir.

Blok zinciri, en yalın hâliyle dijital bir veri kayıt sistemidir. İlkesel olarak merkeziyetsiz yapılarda kullanılmakta ve dağıtık bir yapıya sahiptir. Veriler, zaman damgası ile işlenmiş, birbirine kriptografik olarak bağlanmış bloklar hâlinde kaydedilir ve bu kayıtlar değiştirilemez niteliktedir. Bu mimari yapı, güvenin bir otoriteye değil, matematiksel prensiplere ve ağdaki doğrulama süreçlerine dayandığı bir sistem inşa etmektedir.

Bu yapının anlaşılabilmesi için blok zincirinin tarihsel olarak nasıl şekillendiği ve hangi teknolojik adımların bu sistemin doğmasına öncülük ettiği üzerinde durulması önem arz etmektedir. Özellikle dijital imza, asimetrik şifreleme, özetleme fonksiyonları ve dağıtık zaman damgası gibi gelişmeler, merkeziyetsiz sistemlerin temelini oluşturmuştur. Bu süreçteki temel amaç; bilgilerin yalnızca yetkili kişilerce ulaşılabilir olmasını sağlamak, veri bütünlüğünü güvence altına almak ve bunu herhangi bir merkezi otoriteye gerek kalmadan başarabilmektir.

2008 yılında yaşanan küresel finans krizinin ardından, Satoshi Nakamoto takma adıyla yayımlanan “Bitcoin: A Peer-to-Peer Electronic Cash System” adlı makale, blok zincirinin ilk defa işlevsel bir ürün olarak kullanılmasını sağlamıştır. Bu makalede sunulan yapı; elektronik imza, dağıtık defter ve iş kanıtı gibi kavramlar etrafında, merkeziyetsiz bir ödeme sistemi kurgusunu hayata geçirmiştir. Bitcoin, blok zinciri teknolojisinin yalnızca bir uygulamasıdır. Teknolojinin kendisi ise çok daha geniş potansiyele sahiptir.

Bitcoin’in zamanla bir yatırım aracına dönüşmesi, kripto borsaların ortaya çıkışı ve yüksek getirili işlemlerin cazibesi, blok zinciri teknolojisinin popüleritesini hızla artırmıştır. Ancak

bu pop lerleşme  o unlukla finansal spe ulasyon odaklı olmuŗ, teknolojinin di er potansiyel kullanım alanları g lgede kalmıŗtır. Ethereum platformunun geliŗtirilmesiyle birlikte, blok zinciri sistemlerinin     nc  taraflara gerek olmadan s zleşmelerin y r t lmesini sa layan akıllı s zleşmeler aracılı ıyla  ok daha geniŗ alanlarda kullanılabilece i ortaya konmuŗtur.

Blok zinciri teknolojisinin finans alanındaki ilerlemesini yalnızca bir baŗlangı  olarak g ren, ve asıl potansiyelinin kamu hizmetleri, sa lık sistemleri, tedarik zinciri y netimi gibi finans dıŗı alanlarda kullanılmaya baŗlanmasıyla ortaya  ıkaca ına inanan kiŗi sayısı g n ge tik e artmaktadır. Merkeziyetsiz otonom organizasyonlar (DAO) aracılı ıyla y netim s re lerinin bir otoriteye ihtiya  duymadan y r t lebilmesi, blok zinciri teknolojisinin organizasyonel yapıları da d n ŗt rebilece ini g stermektedir. Bu kapsamda  ok sayıda kamu kurumu ve  zel řirket, blok zincirinin sundu u veri g venli i ve de iŗtirilemezlik  zelliklerini sistemlerine entegre etmeye d n k pilot projeler geliŗtirmektedir.

Ancak bu geliŗmelerin yanı sıra, blok zinciri tabanlı sistemlerin hayatımıza tam anlamıyla entegre olmasının  n nde bir ok yapısal engel bulundu u hususunun g z  n nde bulundurulması faydalı olacaktır. Kripto para hırsızlıkları, g venlik a ıkları, iflas eden borsalar, sistemin y ksek enerji t ketimi,  l eklenebilirlik problemleri, teknik bilgi gereksinimi ve kullanıcı deneyiminin karmaŗıklı ı gibi sorunlar h len   z lmeyi beklemektedir.  zellikle teknolojinin ortalama bir kullanıcının do rudan y netebilece inden  ok daha karmaŗık olması, bu sistemlerin geniŗ kitlelerce benimsenmesini geciktirmektedir. Buna ek olarak, sistemin arkasında bir otorite veya kurumsal g vencenin olmaması, kullanıcıların psikolojik g venlik ihtiya ını karŗılamada yetersiz kalmakta, bu da     nc  kiŗi ve kuruluŗlara yeniden duyulan g ven ihtiya ını artırmaktadır. Bu ba lamda, teorik olarak merkeziyetsiz olan sistemlerin pratikte yeniden merkezileŗmeye baŗladı ı bir ikilem ortaya  ıkmaktadır.

Tezin amacı, yukarıda  er evesi  izilen bu d n ŗ m potansiyelini hem teknik hem de teorik d zeyde de erlendirmek; blok zinciri sistemlerinin matematiksel altyapısını ortaya koymak ve bu altyapıdan hareketle geliŗtirilebilecek uygulamalara katkı sa lamaktır. Bu do rultuda,

sadece popöler tartışmalara deęil, aynı zamanda teknolojinin temel prensiplerine, matematiksel dayanaklarına ve toplumsal yansımalarına da deęinilerek çok yönlü bir akademik çalışma ortaya konulmaya çalışılmıştır.



2. BLOK ZİNCİRİ TEKNOLOJİSİ

Tezimizin temel konusu olan “blockchain” terimine karşılık Türk Dil Kurumu (TDK) tarafından henüz Türkçe bir karşılık üretmediği görülmektedir. Bununla beraber 2018 yılında konu hakkında bir vatandaş tarafından sorulan soruya verilen cevap kapsamında hazırlanan 09.03.2018 tarihli ve 472 sayılı yazıda “blok zinciri” şeklinde kullanımın daha uygun olacağının değerlendirildiği belirtilmektedir. Tezimizde TDK’nın önerisi dikkate alınarak “blockchain” terimi “blok zinciri” olarak kullanılacaktır.

2.1. Blok Zinciri Teknolojisinin Tanımı

Blok zinciri teknolojisinin tek ve net bir terimsel tanımını yapmak oldukça güçtür. Her ne kadar blok zinciri teknolojisi aşama aşama gelişerek Satoshi Nakamoto’nun Bitcoin whitepaper’ı ile günümüzdeki biçimine kavuşmuş olsa da, bu temel makalede "blockchain" (blok zinciri) ifadesi geçmemektedir. Bu kaynakta “chain of bloks” kavramı ile ilgili yapıdan bahsedilmiştir. Ancak söz konusu makaleden sonra Bitcoin sisteminin altyapısını oluşturan teknolojiye "blok zinciri" adı verilmiştir.

Temel olarak blok zinciri teknolojisi, verilerin güvenli ve şeffaf bir biçimde dijital olarak saklandığı ve blokların birbirine kriptografik algoritmalar aracılığıyla bağlandığı, merkezi olmayan dağıtık bir kayıt sistemidir. Farklı kaynaklarda yapılan tanımlara bakıldığında, birbirine benzer ifadelerin kullanıldığı görülmektedir. Genel kabul gören tanımıyla blok zinciri; merkezi olmayan, dağıtık yapıya sahip, herkese açık, sürekli büyüyen ve kalıcı, doğrulanabilir kayıtların bulunduğu bir defterdir. Bu kayıtlar birbirine bağlıdır ve kriptografi sayesinde değiştirilemez hale getirilmiştir. [1]

Blok zinciri teknolojisinin net ve sabit bir tanımının yapılamamasının temel nedeni, farklı projelerde ihtiyaca göre bazı temel özelliklerin esnetilebilmesi veya kısıtlanabilmesidir. Örneğin kamu kuruluşları tarafından yürütülen blok zinciri projelerinde katılımcılar sınırlanabilir ya da verilerin herkes tarafından görüntülenmesi istenmeyebilir. Dolayısıyla merkezi otoritelerin kontrolünde, herkese açık olmayan blok zinciri uygulamaları da

bulunmaktadır. Bu projeler temel blok zinciri tanımındaki bazı özelliklerden feragat etmiş olsalar bile, yine de bir blok zinciri yapısı olarak kabul edilmektedir.

Sonuç olarak, blok zinciri teknolojisi; dijital verilerin bloklar halinde kaydedildiği, bu blokların çeşitli kriptografik yöntemlerle birbirine bağlandığı ve bu sayede veri güvenliğinin sağlandığı teknolojiler bütünü olarak tanımlanabilir. Bu çalışmada mümkün olduğunca yukarıda belirtilen geniş tanım kullanılacaktır.

Blok zinciri türleri temel olarak üç ana kategoriye ayrılır.

Açık (Genel) Blok Zinciri – (Public Blockchain): Herkesin katılabildiği, verileri okuyabildiği ve işlem gerçekleştirebildiği blok zinciridir. Tamamen şeffaf ve merkeziyetsizdir. Genellikle kripto paraların altyapısını oluşturmaktadır. Konsensüs mekanizmaları kullanarak güvenlik sağlanmaktadır. Bitcoin ve Ethereum en bilinen örnekleridir.

Kapalı (Özel) Blok Zinciri – (Private Blockchain): Erişimi yalnızca belirli kişi veya kuruluşlara açık olan, merkezi bir yapı tarafından kontrol edilen blok zinciridir. Erişim ve işlem yetkileri kısıtlıdır. Genellikle şirket içi veri güvenliği ve iş süreçlerinin takibi için kullanılır. Hızlı ve düşük maliyetli işlemler sunar. Hyperledger Fabric ve Corda en bilinen örnekleridir.

İzinli (Konsorsiyum) Blok Zinciri – (Consortium/Federated Blockchain): Birden fazla kurumun birlikte yönettiği yarı-açık blok zinciridir. Katılımcılar önceden belirlenir ve genellikle güvenilir kurumlardır. Kamuya açık değildir ama tamamen kapalı da değildir. Sektörel iş birliği, tedarik zinciri yönetimi gibi alanlarda yaygın kullanılır. IBM Food Trust ve R3 Corda en bilinen örnekleridir.

Tanım karmaşasını önlemek adına iki terimin daha net bir şekilde açıklanması gerekmektedir. Doğaları gereği “Kapalı (Özel) Blok Zincirleri” ve “İzinli (Konsorsiyum)

Blok Zincirleri” yapısal olarak “İzinli (Permissioned) Blok Zinciri” niteliği taşımaktadırlar. Genel (Public) ve Özel (Private) Blok Zinciri yapılarının bir arada kullanıldığı sistemler ise lirectatürde **Hibrit (Hybrid) Blok Zinciri** olarak tanımlanmaktadır.

2.2. Blok Zincirinin Tarihçesi ve Gelişimi

Blok zinciri teknolojisinin daha iyi kavranabilmesi için, bu teknolojiyi mümkün kılan tarihsel gelişmelerin incelenmesi oldukça faydalı olacaktır. Bu kapsamda, kriptografik altyapının evrimi ve dağıtık sistem mantığının tarihsel gelişimi, blok zinciri fikrinin nasıl oluştuğunu anlamak açısından kritiktir. Bu çerçevede, başlangıç noktası olarak dijital ortamda güvenli veri paylaşımını mümkün kılan “dijital imza” kavramının ele alınması yerinde olacağı kanaatindeyiz.

Bu alandaki ilk büyük dönüşümlerden biri, 1976 yılında Diffie ve Hellman tarafından kaleme alınan “New Directions in Cryptography” başlıklı makale [2] ile gerçekleşmiştir. Bu çalışma ile kriptografiye yeni bir yön verilmiş, çift anahtarlı (asimetrik) şifreleme sisteminin temelleri atılmıştır. Simetrik anahtarlarda yaşanan anahtar dağıtım sorununa çözüm sunan bu yaklaşım, daha sonra dijital güvenliğin birçok alanına entegre edilmiştir. Bu sistem sayesinde kişiler, özel anahtarlarını paylaşmaksızın dijital ortamlarda güvenli işlemler yapabilmekte ve bu işlemler başka taraflarca doğrulanabilmektedir.

Asimetrik şifrelemenin pratikte nasıl uygulanabileceğine dair daha ayrıntılı bir yaklaşım, 1977 yılında Rivest, Shamir ve Adleman tarafından sunulan “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems” başlıklı makaleyle [3] getirilmiştir. Yazarların baş harflerinden oluşan RSA algoritması hem dijital imza hem de şifreleme için kullanılabilir bir yöntem olarak halen yaygın biçimde kullanılmaktadır. Günümüzde e-ticaret, e-devlet uygulamaları ve çeşitli güvenlik protokolleri yaygın kullanıma örnek olarak verilebilir.

Blok zinciri fikrine katkı sağlayan erken çalışmalardan biri David Chaum tarafından 1982 yılında yayımlanan “Computer Systems Established, Maintained, and Trusted by Mutually Suspicious Groups” başlıklı yayındır [4]. Bu çalışmada Chaum, güvenin taraflar arasında

dağıtılması yoluyla merkezi otoriteye olan ihtiyacın ortadan kaldırılabilceğini öne sürmektedir. Chaum, daha sonra 1985 yılında anonim dijital kimlik yapısının temellerini attığı “Security Without Identification” [5] isimli çalışması ile mahremiyet temelli dijital sistemlerin öncülerinden biri olmuştur. Bu çalışmalarda sadece veri güvenliği değil, aynı zamanda bireysel gizliliğin korunması da ön plana çıkarılmıştır. 1989 yılında ise Chaum, ilk dijital para örneklerinden biri olan eCash projesini tanıtmıştır. Bu çalışmalar, hem dijital imza hem de merkeziyetsiz yapılar üzerinden blok zinciri teknolojisinin temellerine önemli katkılar sunmuştur.

1991 yılında Stuart Haber ve Scott Stornetta tarafından yayımlanan “How to Time-Stamp a Digital Document” başlıklı makale [6] ise blok zincirinin doğrudan öncülerinden biri olarak kabul edilir. Bu çalışmada yazarlar, dijital belgelerin güvenilirliğini sağlamak amacıyla özetleme fonksiyonlar kullanarak zaman damgası ekleme yöntemini geliştirmiştir. Zaman damgası kavramı, belge üzerinde daha sonra yapılacak herhangi bir değişikliğin algılanabilmesini mümkün kılarak değiştirilemezlik ilkesine katkı sunmuştur. Bu makale, belgelerin değiştirilemezliği ve sürekliliği gibi ilkeler üzerinden bugünkü blok zinciri mantığının temelini atmıştır. Bu makale, Satoshi Nakamoto'nun Bitcoin whitepaper'ında kaynak olarak referans gösterdiği ilk yayındır.

Blok zincirinin önemli bileşenlerinden biri olan konsensüs mekanizmalarının en önemli örneği olan iş kanıtı (Proof of Work) mekanizmasına giden yol, 1992 yılında Dwork ve Naor tarafından yayınlanan “Pricing via Processing or Combatting Junk Mail” isimli makaleye [7] dayanmaktadır. Bu çalışmada, sistem kaynaklarını kötüye kullanan e-posta spam'lerine karşı bir önlem olarak istemcilerle işlem maliyeti getiren kriptografik bir engel önerilmiştir. Kötü niyetli istemcinin belirli bir matematiksel problemi çözmek zorunda kalması işlem yapmayı maliyetli hale getirmektedir. Bu yaklaşım daha sonra 1999 yılında Markus Jakobsson ve Ari Juels tarafından “Proofs of Work and Bread Pudding Protocols” isimli makaleyle [8] sistematik hale getirilerek “Proof of Work” kavramı literatüre kazandırılmıştır. Bu gelişme, spam engellenmenin ötesinde genel anlamda merkeziyetsiz ağlarda dürüst davranışı teşvik eden bir yapı olarak değerlendirilmiştir. 2004 yılında Hal Finney, bu yaklaşımı geliştirerek “Reusable Proof of Work (RPOW)” sistemini öne sürmüş, bu da Bitcoin'in altyapısına doğrudan katkı sağlamıştır.

2005 yılında bilgisayar bilimci ve hukukçu Nick Szabo tarafından ortaya atılan “Bit Gold” kavramı ise, dijital kıtlık ve merkeziyetsizlik ilkeleri üzerine kurgulanmış ilk dijital para birimlerinden biri olarak dikkat çekmektedir. Bu sistemde, kullanıcılar karmaşık hesaplamalar yaparak zaman damgalı zincirler üretebilmekte ve bunları merkezi bir otorite olmaksızın saklayabilmektedir. Szabo’nun çalışması, hem PoW temelli bir değer üretimi hem de kayıtların değiştirilemez biçimde zincirlenmesi gibi Bitcoin’in yapısal öğelerini barındırmaktadır.

2008 yılında ABD’de yaşanan finansal krizin merkez bankaları ve finansal aracı kurumlara olan güveni sarsmasının hemen ardından, Satoshi Nakamoto adını kullanan bir kişi ya da grup tarafından “Bitcoin: A Peer-to-Peer Electronic Cash System” başlıklı makale [9] yayınlanmıştır. Bu makalede daha önce birbirlerinden bağımsız geliştirilen dijital imza, kriptografi, dağıtık defter yapısı, zaman damgası ve iş kanıtı mekanizması bir araya getirilerek kişiler arası aracısız bir ödeme sistemi tanıtılmıştır. Kanaatimizce Satoshi’nin yaklaşımı, sadece teknik bir çözüm değil, aynı zamanda finansal araçlara duyulan güvene karşı felsefi bir duruş olarak da değerlendirilmelidir. Bitcoin, bu bileşenlerin daha önceki çalışmalardan derlenerek bir araya getirildiği, ancak yeni bir teknolojiye çok, mevcut teknolojilerin yenilikçi bir kombinasyonu olarak değerlendirilmelidir.

Burada iki temel noktaya ayrıca vurgu yapmak gerekir. Birincisi, blok zinciri ile Bitcoin kavramlarının sıkça karıştırılmasıdır. Blok zinciri bir altyapı teknolojisi iken, Bitcoin bu altyapı üzerine inşa edilmiş ilk uygulamalardan biridir. İkincisi ise Bitcoin, Satoshi Nakamoto tarafından merkeziyetsiz bir ödeme sistemi olarak tanıtılmıştır. Ancak blok zinciri teknolojisi sadece ödeme sistemleriyle sınırlı kalmamış; akıllı sözleşmeler, merkeziyetsiz organizasyonlar, seçim sistemleri, tapu kayıtları, telif hakları ve veri doğrulama sistemleri gibi pek çok alanda uygulama bulmuştur.

Bitcoin’in ilk blok zinciri işlemi, 12 Ocak 2009 tarihinde Satoshi Nakamoto tarafından Hal Finney’e 10 BTC gönderilmesiyle, blok 170’te [10] gerçekleşmiştir. 2010 yılının Mart ayında ilk Bitcoin borsası olan BitcoinMarket.com kurulmuş ve Bitcoin ilk kez dolar karşılığında alınıp satılabilir hale gelmiştir. Aynı yılın Mayıs ayında ise fiziksel karşılığı olan ilk Bitcoin

işlemi gerçekleşmiş ve 10.000 BTC karşılığında pizza siparişi verilmiştir. Bu gelişmeler, Bitcoin'in sadece teknik değil, aynı zamanda ekonomik bir fenomen olarak da yaygınlaşmasını sağlamıştır. Aşağıda yer alan şekilde Bitcoin'in zaman için de fiyat değişimi Dolar (USD) cinsinden gösterilmektedir.

Şekil 1: Bitcoin Fiyat (USD) Zaman Grafiği



Grafik www.tr.tradeview.com sitesinden 21.11.2024 tarihinde alınmıştır.

Bitcoin'in bilinirliğinin artması ile birlikte blok zinciri teknolojisi ile neler yapılabileceği üzerine projeler de ortaya çıkmaya başlamıştır. 2011 yılında Namecoin internet alan adı sistemini tanıtmıştır. Namecoin, merkezi olmayan bir DNS sistemi kurarak internet alan adlarının sansürden korunmasını amaçlamıştır. Bitcoinin hemen birkaç yıl sonrasında daha hızlı olmayı hedefleyen yeni coin projeleri hayata geçirilmiştir. Bu dönemde Litecoin, blok süresini kısaltarak işlem onaylarını hızlandırmayı amaçlayan ilk coinlerden biri olmuştur. Ripple geleneksel bankacılık sistemine bir alternatif olmak üzere merkezi bir proje olarak dikkatleri üzerine toplamıştır. Özellikle sınır ötesi para transferlerinde düşük maliyet ve hız avantajı sağlayan Ripple, birçok banka ile iş birliği geliştirmiştir. Monero blok zinciri felsefesinin şeffaflık ve izlenebilirlik kurallarını tamamen ortadan kaldıran gerçek gizlilik sağlayan bir coin olarak ekosisteme dahil olmuştur. Monero'nun sağladığı gizlilik, ring signature ve gizli adresler (stealth address) gibi teknolojilere dayanmaktadır. Ethereum ile blok zinciri başka bir seviyeye taşınmış ve artık akıllı kontratlar, merkezi olmayan finans

uygulamaları ve yönetim modelleri konuşulmaya başlanmıştır. Vitalik Buterin'in liderliğinde geliştirilen Ethereum, programlanabilir blok zinciri altyapısı sunarak yalnızca para transferi değil, birçok uygulamanın zincir üzerinde çalışmasını mümkün kılmıştır. Bu süreçte birçok farklı teknoloji ve yenilik getiren onlarca coin ekosisteme dahil olmuştur. Bunlar arasında DAO yapıları, DeFi protokolleri, NFT platformları ve Layer 2 çözümleri gibi bileşenler zamanla öne çıkmıştır.

Bu tarihi ilerlemelerle birlikte blok zinciri ekosisteminin bazı sorunları da göz önüne çıkmaya başlamıştır. Bunların başında, artan kullanıcı ve işlem sayısı ile birlikte ortaya çıkan ölçeklenebilirlik sorunu gelmektedir. Ölçeklenebilme problemi birçok eski coinin yavaş ve pahalı kalmasına sebep olmuş; bu sorunu çözmek üzere de ikinci katman çözümleri (Layer 2) ve alternatif blok zinciri mimarileri ortaya çıkmıştır. Özellikle Lightning Network (Bitcoin) ve Optimistic/ZK Rollup (Ethereum) gibi yapılar, işlem sayısını artırarak işlem ücretlerini düşürmeyi hedeflemiştir.

Çok eleştirilen bir başka husus da elektrik tüketiminin fazlalığı konusudur. Özellikle Proof of Work (PoW) sistemlerinde madencilik faaliyetleri yüksek enerji gereksinimi doğurmuştur. Blok zinciri konusuna yabancı olan insanların en abartıldığını düşündüğü konu kanaatimizce elektrik tüketimi konusudur. Bilgisayara indirilen bir uygulamanın sürekli internete bağlı olması ve çalışması ile harcanacak elektriğin bu kadar sorun olmayacağı doğal bir hissiyat olsa da; sistemin mimarisi gereği kullanıcı sayısı arttıkça yapılacak işlemin zorluğunun artması, dolayısıyla gerekli işlem gücünün artması bir kısır döngü yaratmaktadır. Bu döngü, “madencilik yarışına” dönüşerek giderek daha güçlü ve enerji yoğun donanımların kullanılmasını zorunlu kılmıştır. Bir özetleme fonksiyon değerini hesaplamak üzere binlerce makine çalışmakta ve sadece biri tarafından bu değerinin bulunması sonrası yeni özetleme fonksiyonu değeri için yarış tekrar başlamaktadır. Bitcoin'in değerinin artması ile madencilğe olan talep daha da artarak, bilgisayarların CPU ve GPU özetleme fonksiyonu hesaplama işlem kapasitesinden binlerce kat daha güçlü yeni donanımlar üretilmesini teşvik etmiştir. Bu donanımlardan binlercesinin bulunduğu madencilik çiftlikler kurularak madencilik farklı bir seviyeye taşınmıştır. Bu bağlamda, Bitcoin madenciliği çevresel etkileri nedeniyle birçok eleştiriye maruz kalmaktadır. T.C. Enerji ve Tabii Kaynaklar Bakanlığı'nın rakamlarına göre 2024 yılında Ülkemizin yıllık elektrik tüketimi

347,9 TWh olmuştur. Sadece Bitcoin madenciliği sonrası 2024 yılında tüketilen elektrik ise Cambridge Centre for Alternative Finance (CCAF) tarafından yayınlanan Cambridge Bitcoin Electricity Consumption Index (CBECI)'e göre yaklaşık 170 TWh olarak gerçekleşmiştir. Bu miktar, tek başına birçok ülkenin toplam elektrik tüketimine eşdeğer seviyededir. Bu çok ciddi bir tüketimdir ve parasal değeri yaklaşık 27 milyar USD'dir. Bu durum, çevresel sürdürülebilirlik açısından kripto para topluluklarını ve geliştiricilerini yeni çözüm arayışlarına yönlendirmiştir. Blok zinciri teknolojisinde elektrik tüketimine karşı yeni madencilik mekanizmaları geliştirilmiş ve yeni birçok coin farklı mekanizmalar kullanmaya başlamıştır.

Yukarıda bahsettiğimiz bazı gelişmeler doğrultusunda tabloya genel olarak bakıldığında, yıllar içerisinde birçok yeni teknolojinin ortaya çıktığı, bu teknolojilerin karşılaştığı sorunlara çözümler üretildiği ve blok zinciri ekosisteminin belli bir doyumluğa geldiği görülmektedir. 2020'li yıllara gelindiğinde blok zinciri teknolojisi, yenilikten çok, mevcut yapılar üzerine geliştirme ve entegrasyon süreçleriyle gündeme gelmeye başlamıştır. Son yıllarda giderek blok zinciri haberlerinin azaldığı ve yeni projelerin dikkat çekici çıkışlar yapmadığı görülmektedir. Bunun en önemli nedenlerinin; bu teknolojinin belli bir doyumluğa ulaşmış olması, insanlar tarafından kabul görmesi için bir zamana ihtiyaç duyması ile yasal düzenlemelerin yavaş ve temkinli olması olduğu kanaatindeyiz. Birçok ülkede blok zinciri uygulamaları hâlen gri alanlarda faaliyet göstermekte, bu da projelerin ölçeklenmesini ve geleneksel yapılarla entegrasyonunu zorlaştırmaktadır. Ancak düzenleyici belirsizliklerin zamanla aşılması ve teknik altyapının daha kullanıcı dostu hale gelmesiyle birlikte blok zinciri tabanlı uygulamaların daha yaygın hale gelmesi mümkün olacaktır. Otoritelere karşı bir duruş olan bu teknolojinin otoriteler tarafından hemen kabul edilmesini beklemenin gerçekçi olmadığı aşikardır.

Bu kapsamda günümüzde teknolojik yenilik denildiğinde yapay zekâ haberlerinin daha baskın olduğu ve önemli yatırım fonlarının bu alana yönlendiği görülmektedir. Yapay zekâ modellerinin günlük hayatı kolaylaştıran araçlara dönüşmesi, kullanıcı ilgisinin bu yöne kaymasına neden olmuştur. Bu durumun blok zinciri teknolojisinin bir süre daha yatay seyretmesine sebep olacağı kanaatindeyiz. Ancak yapay zekanın insanların hayatına daha fazla girmesi ve kişiye özel yapay zekâ asistanlarının kullanılmaya başlanması ile blok

zincirinin yeniden eski popülaritesine kavuşacağı düşünülmektedir. Özellikle yapay zekaya dayalı kişisel verilerin güvenli ve denetlenebilir biçimde saklanması için blok zinciri altyapısının kritik bir rol üstlenmesi beklenmektedir.

2.2.1. Blok Zinciri Hakkında Dikkat Çekici Olaylar ve Kritik Vakalar

Blok zinciri teknolojisinin gelişim süreci, yalnızca teknik ilerlemelerle değil, aynı zamanda kamuoyunun ilgisini çeken olaylar ve dönüm noktalarıyla da şekillenmiştir. Geliştirici topluluğun oluşumu, erken dönem işlemler, bilinmeyen kimlikler ve yüksek profilli siber saldırılar gibi çeşitli olaylar, hem sistemin evrimini hem de toplumsal algısını etkilemiştir. Bu bölümde, blok zinciri tarihine damga vuran teknik, kültürel ve güvenlik odaklı bazı dikkat çekici gelişmelere yer verilecektir.

Dikkat çekici en önemli detay Satoshi Nakamoto kimliğinin hala bilinmemesidir. Birçok kişi Bitcoin'in bir kişi tarafından oluşturulamayacak kadar karmaşık olduğunu iddia etmektedir. Bazı araştırmacılar ise bu teknolojinin arkasında akademik bir ekip ya da devlet destekli bir grup olabileceğini öne sürmektedir. Bazı insanlar ise bu teknolojinin gizli servisler ya da bazı devletler tarafından test amaçlı kullanıma sunulduğunu iddia etmektedirler. Günümüzde Bitcoin fiyatlarına ve Satoshinin sahip olduğu Bitcoin sayısına bakıldığında ortadan kaybolması ve bu serveti atıl bırakması çok anlaşılabilir değildir. Öyle ki, tahmini 1 milyon BTC'ye erişebilecek büyüklükte bir varlık, yaklaşık onbeş yıldan uzun süredir hiç hareket ettirilmemiştir.

Çalışmalarımız sırasında, Satoshi'nin sürecin başlarında Bitcoin forumlarında teknik konularda tartışmalara katıldığı, bazı konularda detaylı teknik bilgi verdiği ve kullanıcı önerilerine karşı duyarlı olduğu görülmektedir. Bilinen son halka açık mesajı, 12 Aralık 2010 tarihinde Bitcointalk forumunda yayınlanmıştır. 2011 yılında Bitcoin geliştiricisi Martti Malmi'ye gönderdiği özel bir e-postada, “Başka şeylere geçtim. Muhtemelen bundan sonra ortalıkta olmayacağım.” şeklinde bir ifade kullanmıştır. Bazı gazete haberlerine karşılık olarak 2014 yılında P2P Foundation forumunda “Ben Dorian Nakamoto değilim” şeklinde kısa bir açıklama yayımlamıştır. Bu, Satoshi'ye ait olduğu son kez doğrulanabilen ifadedir.

İnternet sayfası üzerinden yaptığımız incelemede bitcoing.org domain adının 18 Ağustos 2008 tarihinde kimlik gizliliği sağlayan bir anonim şirketten alındığı görülmektedir [11]. 2009 yılının Ocak ayında Bitcoin'in ilk sürümü (v0.1.0) yayımlanmıştır. Satoshi Nakamoto'nun Windows tabanlı ilk sürümünü paylaşmasının ardından, Martti Malmi projeye katılmış ve Linux uyumluluğunu geliştirerek erişimi küreselleştirmiştir. Zamanla bu sitenin yönetimi Bitcoin topluluğuna devredilmiştir. Martti Malmi, Satoshi ile sadece e-posta yoluyla iletişim kurduğunu ve kimliğini hiç sormadığını belirtmiştir. Sitenin finansmanı ve Bitcoin kod geliştirmeleri artık bu topluluk tarafından yapılmaktadır.

Satoshi Nakamoto'nun açık kaynak kodu paylaşp yaklaşık 2 yıl geliştirilmesi için çaba harcadıktan sonra ortadan kaybolduğu bilinmektedir. Daha sonra bilinen başlıca geliştiriciler Gavin Andresen ve Wladimir J. van der Laan olmuştur. Günümüzde Satoshi Nakamoto'nun kim olduğu hala tartışma konusudur.

Martti Malmi, Bitcoin'in ilk kez bireysel düzeyde fiat para karşılığı takasını gerçekleştirmiştir. Bu işlem 2009 yılında, PayPal üzerinden 5.050 BTC'yi 5 USD karşılığında satmasıyla gerçekleşmiştir. Ancak Bitcoin'in piyasa düzeyinde dolar karşılığı işlem görmesi, 2010 yılında açılan bitcoinmarket.com borsası üzerinden mümkün olmuştur.

Blok zinciri ve kripto paralar konusunu araştırırken en fazla karşılaşılan kaynak site GitHub olacaktır. Açık kaynak kodlu ve geliştirilmeye açık projeler genellikle uzmanlaşmış platformlar üzerinden paylaşılmaktadır. GitHub, sadece bir dosya barındırma yeri değil, aynı zamanda yazılımcıların işbirliği yapabildiği, katkı takibi yapıldığı ve projelerin şeffaf biçimde geliştirilebildiği bir platformdur. Bu kapsamda Bitcoinin ilk versiyonu o dönem popüler olan SourceForge platformunda yayımlanmıştır. Ücretsiz barındırma hizmeti vermesi, proje için hem yönetim hem de dağıtım kolaylığı sağlaması açısından bu platformlar tercih edilmekteydi. SourceForge üzerinde Satoshi'nin yüklediği ilk versiyon (v0.1.0), dönemin açık kaynak yazılım kültürünü yansıtmaktadır. 2011 yılında projelere katkı sağlamayı kolaylaştıran ve daha kapsamlı kontrol hizmeti sunan GitHub platformuna geçilmiştir. Bu geçiş, sadece yazılım geliştirme değil, aynı zamanda geliştirici topluluğunun büyümesi açısından da önemli bir dönüm noktasıdır.

Bir başka dikkat çekici olay da 10.000 Bitcoine alınan pizza olayıdır. Bitcoin geliştiricilerinden biri olan ve Florida, ABD’de yaşayan Laszlo Hanyecz 22 Mayıs 2010 tarihinde bitcointalk formunda “10.000 Bitcoin karşılığında iki büyük boy pizza alabilecek birini arıyorum.” yazmış, yine bu form kullanıcısı Jeremy Sturdivant teklifi kabul ederek kendi kredi kartıyla sipariş vererek Laszlo Hanyecz’in adresine pizza göndermiştir. Karşılığında da o gün 41 dolar değerinde olan 10.000 bitcoin almıştır. Bitcoinle yapılan ve fiziksel karşılığı olan ilk işlem olması sebebiyle önemli kabul edilen bir olaydır. 22 Mayıs günü “Bitcoin Pizza Günü” olarak kutlanmaktadır.

NFT (Non-Fungible Token) piyasası, özellikle 2021 yılında dijital sanat eserlerine yönelik yoğun ilgiyle büyük bir ivme kazanmıştır. Bu dönemde gerçekleşen en çarpıcı olaylardan biri, dijital sanatçı Beeple takma adlı Mike Winkelmann’ın oluşturduğu “Everydays: The First 5000 Days” adlı eserinin açık artırmada 69,3 milyon dolara alıcı bulmasıdır. Eser, sanatçının yaklaşık 13 yıl boyunca her gün ürettiği dijital görsellerin bir araya getirilmesinden oluşmaktadır. 11 Mart 2021’de Christie’s tarafından düzenlenen bu müzayede, NFT’lerin sadece teknolojik değil, aynı zamanda kültürel ve ekonomik bir fenomen haline geldiğinin somut bir göstergesi olmuştur. [12]

Mt. Gox 2014 yılında dünyanın en büyük Bitcoin borsasından biri olarak bilinmekteydi. Ancak, Şubat 2014’te borsa, yaklaşık 850,000 Bitcoin’in (tüm Bitcoin miktarının %6’sı) kaybolduğunu açıkladı. Bu olay Borsanın iflasına ve Bitcoin fiyatlarında büyük düşüşlere yol açmıştır. Yaklaşık 460 milyon dolar değerinde bitcoin kaybolmuştur. Daha sonra 200.000 BTC’nin borsa tarafından kurtarıldığı açıklanmış; geri kalan kısmın büyük ölçüde çalındığı tahmin edilmiştir. Olayın ardından Şirket CEO’su Mark Karpeles tutuklanmıştır. Kayıp sebepleri hırsızlık ve kötü yönetim olarak belirtilmiştir [13].

The DAO, Ethereum blockchain üzerinde oluşturulmuş merkezi olmayan bir otonom organizasyon olarak kurulmuştur. Haziran 2016’da bir saldırıya uğramış ve yaklaşık 50 milyon dolar (3.6 milyon Ether) çalınmıştır. Bu saldırı, The DAO’nun akıllı sözleşme kodundaki bir güvenlik açığından kaynaklanmıştır. Bu olay, Ethereum topluluğu içinde tartışmalara sebep olmuştur. Topluluk, fonları geri alabilmek adına Ethereum ağında bir

“hard fork” (sert çatallanma) yapılmasına karar vermiştir. Bu karar sonucunda Ethereum ikiye ayrılmıştır: saldırı öncesi fonları geri alan yeni zincir “Ethereum (ETH)” ve orijinal zincir “Ethereum Classic (ETC)” olarak yoluna devam etmiştir. [14].

Coincheck, 2018 yılında Japonya merkezli bir kripto para borsasıdır. Ocak 2018'de borsa, yaklaşık 530 milyon dolar değerinde NEM tokenlerinin (XEM) kaybolduğunu açıklamıştır. Siber saldırı olayı, borsanın sıcak cüzdanlarını (hot wallets) hedef alarak gerçekleştirilmiştir. Sıcak cüzdanların yeterince çoklu imza (multi-signature) desteği olmadan çalıştırılması, sistemin zayıf noktasını oluşturmuştur. Olayın ardından Coincheck, yatırımcılarının kayıplarını kısmen geri ödeme kararı almış ve güvenlik protokollerini güçlendirmiştir [13].

Poly Network, merkeziyetsiz finans (DeFi) platformlarından biridir. Ağustos 2021'de, platform bir saldırıya uğramış ve yaklaşık 610 milyon dolar değerinde kripto varlık çalınmıştır. Saldırı sonrası siber saldırganlar, çaldıkları fonları iade etmeye karar vermiş ve bir kısmını geri göndermişlerdir. Bu olay, DeFi protokollerinin güvenlik açıklarını gözler önüne sermiştir. Olayın ardından güvenlik protokolleri iyileştirilmiştir [13].

Ronin Network, popüler oyun Axie Infinity'nin arkasındaki Ethereum yan zinciri olarak bilinmektedir. Mart 2022'de, ağ saldırıya uğramış ve yaklaşık 614 milyon dolar değerinde 173,600 Ethereum ve 25.5 milyon USDC çalınmıştır. Saldırının, validator node'larının sosyal mühendislikle ele geçirilmesiyle gerçekleştiği açıklanmıştır. Bu hırsızlık, oyun sektöründeki blok zincir projelerinin güvenliğinin sorgulanmasına sebep olmuş ve önemli kayıplar olabileceğini göstermiştir [15].

Wormhole, farklı blok zincirler arasında köprü görevi gören bir platformdur. Şubat 2022'de saldırıya uğramış ve yaklaşık 320 milyon dolar değerinde 120,000 Wrapped Ethereum (WETH) çalınmıştır. Saldırı, zincirler arası köprülerin güvenlik açıklarını ortaya koymuş; protokolün akıllı sözleşme doğrulamasındaki bir hata kullanılmıştır. Bu olay, blok zinciri köprülerinin güvenliği konusundaki endişeleri artırmış ve köprü protokollerinin güvenlik önlemlerinin gözden geçirilmesine neden olmuştur [16].

FTX, popüler bir kripto para borsasıdır. Kasım 2022'de, borsa büyük bir saldırıya uğramış ve yaklaşık 600 milyon dolar değerinde kripto varlık çalınmıştır. Şirket CEO'su Sam Bankman-Fried tutuklanmıştır. Olayın ardından borsa iflas niteliğinde mülkiyetten feragat etmiş ve yatırımcılarının kayıplarını kısmen geri ödeme kararı almıştır [17].

Mixin Network, kullanıcıların sıcak cüzdanlarını hedef alan bir saldırıya mağruz kalmıştır. Yaklaşık 200 milyon dolar değerinde kripto varlık kaybolmuştur. Saldırının, merkezi sunucuların siber saldırıya uğramasıyla gerçekleştiği ve güvenlik açıklarının kötüye kullanıldığı bildirilmiştir. Bu saldırı, sıcak cüzdanların güvenliğine dair endişeleri yeniden gündeme getirmiş ve kullanıcıların güvenlik önlemlerini gözden geçirmelerine neden olmuştur [18].

Ocak 2023'te, Multichain platformunu hedef alan saldırılar sonucu kullanıcılar yaklaşık 200 milyon dolar değerinde kripto varlık kaybetmiştir. Bu olay, çapraz zincir (cross-chain) protokollerinin güvenliği konusundaki endişeleri artırmıştır. Benzer şekilde, Euler Finance protokolü, Mart 2023'te 197 milyon dolar değerinde kripto varlık kaybetmiştir. Bu kayba sebep olan saldırı, "flash loan attack" olarak bilinen bir DeFi güvenlik açığından kaynaklanmıştır. Protokol geliştiricileri güvenlik önlemlerini artıracak yeni yöntemler bulmak zorunda kalmıştır.

Şimdiye kadar bilinen en büyük çaplı siber saldırı olayı ise Dubai merkezli Bybit borsasında 21 Şubat 2025 tarihinde gerçekleştirilen 1,5 milyar dolarlık saldırıdır. 400.000'den fazla ETH'nin çalındığı duyurulmuştur. Bu olayı farklı kılan sadece çalınan tutar değil, saldırının yöntemidir. Genellikle kripto siber saldırılarında köprüler, akıllı sözleşmeler ya da kod hataları gibi teknik açıklar hedef alınırken, bu olayda hem sıcak hem de soğuk cüzdanların aynı anda ele geçirilmesi, sektörde büyük bir şok etkisi yaratmıştır. Saldırının yöntemi hâlâ tam olarak açıklanamamış olup, güvenlik uzmanları olası iç tehdit, sosyal mühendislik ve gelişmiş saldırı teknikleri üzerinde durmaktadır [19].

Yukarıda özetlenen bu saldırılar dışında daha birçok siber saldırı olayı yaşanmıştır. Burada bahsettiğimiz olayları özetleyen tablo aşağıda yer almaktadır.

Tablo 1: Blok zinciri önemli siber saldırı olayları

Tarih	Şirket / Platform	Kayıp Tutarı (USD)	Saldırı Yöntemi / Güvenlik Açığı
Şubat 2014	Mt. Gox	460 milyon	Kötü yönetim ve hırsızlık (detaylı yöntem belirsiz)
Haziran 2016	The DAO (Ethereum)	50 milyon (3.6M ETH)	Akıllı sözleşme açığı
Ocak 2018	Coincheck	530 milyon (XEM)	Yetersiz güvenlik sebebiyle sıcak cüzdanların siber saldırıya uğraması
Ağustos 2021	Poly Network	610 milyon	Akıllı sözleşme zafiyeti, çapraz zincir protokol açığı
Şubat 2022	Wormhole	320 milyon (120,000 WETH)	Zincirler arası köprü açığı
Mart 2022	Ronin Network	614 milyon	Validator node ele geçirilmesi (sosyal mühendislik şüphesi)
Kasım 2022	FTX	600 milyon	Şirket içi güvenlik zaafı / iç tehdit (saldırının kaynağı net değil)
Ocak 2023	Multichain	200 milyon	Cross-chain protokol açığı
Mart 2023	Euler Finance	197 milyon	Akıllı sözleşme istismarı (flash loan attack)
Eylül 2023	Mixin Network	200 milyon	Sıcak cüzdan sunucusuna saldırı
Şubat 2025	Bybit	1.5 milyar (<400,000 ETH)	Henüz tam olarak anlaşılamayan yöntem – sıcak ve soğuk cüzdanlar ele geçirildi

2.3. Blok Zinciri Teknolojisinin Felsefesi ve Temel Kavramları

Blok zinciri teknolojisi, yalnızca bir yazılım ya da veri saklama yöntemi değil; aynı zamanda toplumsal ve ekonomik ilişkilerin kökten yeniden kurgulanmasını mümkün kılacak bir yaklaşımdır. Bu sistem, insan doğasının temel bir yönü olan “güven” ihtiyacını, merkezi otoritelere ihtiyaç duymadan karşılamayı hedeflerken güveni bir merkeze değil, topluluğun kolektif karar alma süreçlerine emanet etmektedir.

Blok zincirinin arkasında yatan felsefe; özgürlük, şeffaflık ve adalet ilkelerini bir araya getirmektedir. Özgürlük, bireyin kendi varlığı üzerinde tam söz sahibi olması anlamına gelir. Geleneksel sistemlerde bireyler, finansal ya da sosyal alanlarda merkezi yapıların kontrolü altındayken, blok zinciri bu gücü bireye iade etmeyi amaçlamaktadır. Bu durum, yalnızca

ekonomik yaşamda değil, aynı zamanda toplumsal ve politik düzlemde de bireyin etkisini artırmaktadır.

Şeffaflık, bu teknolojinin en temel taşlarından biridir. Yapılan her işlem, alınan her karar, değiştirilemez biçimde kayıt altına alınmakta ve herkesin erişimine açık hâle getirilmektedir. Bu durum yalnızca yolsuzluğun önlenmesine değil, aynı zamanda toplumsal bilincin gelişmesine de katkı sunmaktadır.

Adalet sistemin tamamının ürettiği bir değerdir. Gücün tek bir elde toplanması, çoğu zaman adaletsizlikle sonuçlanmaktadır. Blok zinciri, bu gücü dağıtarak, bireylerin eşit söz hakkına sahip olabileceği bir yapı sunarken, ekonomik, sosyal ve politik alanlarda daha dengeli bir katılım mümkün hâle gelmektedir.

Ayrıca bu teknoloji, zamanla kurulan güven kavramını da dönüştürmektedir. Geçmişte güven, yıllara dayanan ilişkiler ve yerleşik kurallar üzerinden inşa edilmekteydi. Oysa blok zinciri, bu güveni anlık ve geri döndürülemez biçimde sağlayabilmektedir. Her blok, geçmişin doğruluğunu kriptografik yöntemlerle güvence altına alırken, geleceğin de açık ve izlenebilir biçimde oluşturulacağına dair güven sunmaktadır.

Özünde bu teknoloji, insan doğasının daha yüksek bir bilinç düzeyine erişebilmesini sağlayacak bir araç olma potansiyeline sahiptir. Bireyler arası doğrudan, adil ve güvene dayalı ilişkileri mümkün kılarak yalnızca bir ekonomik devrim değil, aynı zamanda toplumsal bir evrim olacaktır.

Blok zinciri, gücün ve kontrolün bireylerin ellerine geçtiği; toplulukların kendi geleceklerini şekillendirebildiği bir dünya teklif etmektedir. Bu, sadece teknik bir başarı değil; insanlığın kendi potansiyelini gerçekleştirme yolunda attığı ciddi bir adımdır. Bireyi güçlendirirken, toplumsal yapıları da daha adil ve demokratik bir biçimde yeniden inşa etme fırsatı sunmaktadır.

2.3.1. Merkeziyetsizlik (Decentralization)

Merkeziyetsizlik, blok zinciri teknolojisinin en önemli kavramlarından biridir. Bitcoin'in whitepaper'ında da bir otoriteye bağılı olmadan çalışan bir ödeme sistemine vurgu yapılmaktadır. Bir otoriteye bağılı olmamak, doğal olarak, katılımcıların sınırlandırılmadığı, işlemlerin engellenemediğı ve bilgilerin değiştirilemediğı bir ortamın tesis edilmesini gerektirmektedir. Aksi halde, bu işlemlerin yürütülebilmesi için yine bir merkezi yapıya ihtiyaç duyulacaktır.

Merkezi bir otoriteye gereksinim duyulmaması için, onun sağladığı fonksiyonların yerini alacak alternatif bir yapı oluşturulmalıdır. Bu noktada en kritik başlıklar; bilginin güvenliğı, güvenilirliğı ve ulaşılabilirliğı olarak öne çıkmaktadır. Sistem bu üç unsuru kendi içinde sağlayabiliyorsa, dışarıdan bir kontrol ya da denetime ihtiyaç duymadan çalışabilir hâle gelmektedir. Nitekim Bitcoin örneğinde de Satoshi Nakamoto'nun başlattığı sistem, artık hiçbir kişi ya da kurumun yönetimine gerek kalmaksızın işlemeye devam etmektedir. Sistemi ayakta tutanlar ise, bu yapıya inanıp kendi bilgisayarlarına bu yazılımı kurarak düğüm (node) olan kullanıcılarıdır. Bu düğümler arasında çalışan algoritmalar sayesinde işlemler doğrulanmakta, kararlar alınmakta ve sistem kendi içinde devam etmektedir.

Blok zinciri felsefesinden bakıldığında, yukarıdaki teorik açıklamalar önemli bir çerçeve sunar. Ancak meseleye biraz da gerçek dünya uygulamaları üzerinden bakılmasının faydalı olacağı düşünülmektedir. Özellikle sistemin geliştirilmesi ve yönetilmesi açısından bu konunun tartışmaya açık yönleri bulunmaktadır. Bitcoin örneğı ele alındığında sistem bir kez çalışmaya başladıktan sonra dışarıdan herhangi bir müdahaleye ihtiyaç duymamış ve bir yöneticisi de olmamıştır. Fakat zaman zaman yapılan güncellemeler ve yazılım yamaları konusunda geliştirici grubun etkisi dikkat çekmektedir. Genellikle bu işlerin Bitcoin Core ekibi tarafından yürütüldüğü ifade ediliyor olsa da bu güncellemeleri kimlerin hangi yetkilerle yaptığı, kullanıcı adı ve erişim bilgilerine kimin sahip olduğu tam olarak paylaşılmamaktadır. Ayrıca bu kararların alındığı platformlarda yer alan kişilerin çoğunun gerçek kimlikleri bilinmemekte, yalnızca kullanıcı adlarıyla tanınmaktadırlar. İsimleri bilinse bile, bu kişilerin yazılımın yönünü belirleyecek kararları alma gücüne sahip olduğu düşüncesi, sistemin felsefi anlamda merkeziyetsizliğini tartışmalı hale getirmektedir.

Benzer bir durum Ethereum için de geçerlidir. Ethereum'da en büyük stake sahipleri genellikle büyük borsalardır. Bu durum, sistemdeki karar alma mekanizmalarında önemli bir ağırlık oluşturmaktadır. Madencilik ya da stake gücünün belli gruplarda toplanması, teoride hedeflenen dağıtık yapının pratikte zayıflamasına neden olmaktadır.

Yani özetle, teoride merkeziyetsizlik hedeflense de, uygulamada bazı açılardan bu hedefin tam anlamıyla sağlanamamakta olduğu değerlendirilebilir.

Mevcut finans sistemine baktığımızda, giriş ve kullanım hakkı konusunda birçok sınırlama ve ayırım göze çarpmaktadır. Bazı bireyler, kamu otoriteleri tarafından sistem dışına itilebilir, işlem geçmişleri erişilemez olabilir veya belirli işlemler için izin gerekebilir. Bu yönüyle blok zinciri, daha kapsayıcı ve eşitlikçi bir yapı sunma iddiasındadır. Teorik olarak herhangi bir kişinin bu ağlara katılmasının önünde bir engel bulunmamaktadır.

Blok zinciri sisteminde merkezi bir otorite bulunmadığından, sisteme katılan tüm düğümler eşit haklara sahip varlıklar olarak düşünülür. Tek başına karar verme yetkisine sahip olmayan bu düğümler, işlemlerin onaylanması ve kabul edilmesi süreçlerinde belirli algoritmalar doğrultusunda oy çokluğuyla hareket ederler. Bu yapı içerisinde hiçbir düğüm diğerinden üstün değildir; hepsi teorik olarak eşit haklara sahiptir. Ancak bu teorik eşitlik, gerçek dünya uygulamalarında ciddi biçimde aşınmaktadır.

Örneğin Bitcoin'de madencilik havuzları ile tek başına madencilik yapmak isteyen bir kullanıcının imkânları arasında çok büyük fark vardır. Bu fark, madencilik havuzlarının bir tür oligopol piyasası yaratmasına neden olmuştur. Ethereum'da ise stake tabanlı yapıda, 32 ETH'ye sahip olanlar düğüm olabilmekte ve madencilik yapabilmektedir. Bu ciddi bir ekonomik bariyerdir. Solana örneğinde de sistemin gerektirdiği donanım oldukça yüksek düzeydedir. Tüm bunlar, teknik ve ekonomik sınırlamalar nedeniyle pratikte tam anlamıyla bir eşitlik sağlanamadığını göstermektedir. Aşağıda bazı blok zincirlerinde merkeziyetsizlik düzeyleri gösterilmektedir.

Tablo 2: Bazı blok zinciri teknolojilerinin merkeziyetsizlik düzeyleri

Kripto Para	Merkeziyetsizlik Seviyesi	Açıklama
Bitcoin	Yüksek	Oldukça merkeziyetsizdir; ancak büyük madencilik havuzları zaman zaman ağın çoğunluğunu kontrol edebilir.
Monero	Yüksek	Mahremiyet odaklı ve oldukça merkeziyetsizdir, madencilik CPU dostudur ve node yapısı yaygındır.
Uniswap	Yüksek	Tamamen merkeziyetsiz borsa protokolüdür. Kullanıcılar yönetime katılabilir, ancak UNI token sahipleri yönetimde belirleyicidir.
Maker	Yüksek	Dai stabilcoin'i yöneten DAO yapısına sahiptir; yönetim token sahiplerinin oyları kararları belirler.
Ethereum	Yüksek	Proof of Stake geçişi sonrası daha az enerji tüketmektedir. Fakat yüksek miktarda ETH stake edenler ağı yönetiminde daha etkili olmaktadır.
Polkadot	Orta	Ağ yönetimi gelişmiş ve zincirler arası birlikte çalışabilirlik sağlamaktadır. Ancak kurucu ekip hâlâ önemli bir etkiye sahiptir.
Avalanche	Orta	Hızlı ve modüler yapı sunar; ancak Avalanche Foundation'ın ve AVAX dağılımının etkisi merkeziyet riski yaratmaktadır.
Cardano	Orta	Bilimsel yaklaşımıyla öne çıkar, node dağılımı iyidir ancak merkezi geliştirme ekibi önemli rol oynar.
Chainlink	Orta	Gerçek dünya verilerini blokzincire taşıyan oracle ağıdır, ancak node sağlayıcıları sınırlıdır, bu da merkezileşmeye neden olabilir.
Cosmos	Orta	Zincirler arası birlikte çalışabilirlik sağlar, her ağ kendi kurallarını belirleyebilir; stake oranları merkeziyet riski yaratabilir.
Filecoin	Orta	Merkeziyetsiz veri depolama sunar; ancak büyük sağlayıcılar ağı domine edebilir.
Zcash	Orta	Gizlilik sunar, fakat 'trusted setup' süreci ve ana ekip etkisi merkeziyet tartışmalarına yol açar.
Tezos	Orta	Kendi kendini yükseltebilen blokzincirdir, yönetim doğrudan zincir üzerinde yapılır; ancak stake dağılımı eşit değildir.
Solana	Düşük	Yüksek işlem hızı sunar, fakat doğrulayıcı sayısı az ve donanım gereksinimi yüksek olduğundan merkeziyet riski taşımaktadır.
Optimism	Düşük	Ethereum Layer 2 çözümüdür, işlemler merkezi bir sıralayıcı tarafından yapılır, bu da merkezileşmeye yol açar.
Arbitrum	Düşük	Optimism'e benzer şekilde merkezi bir sıralayıcıya sahiptir, ancak gelecekte daha fazla merkeziyetsizlik planlanmaktadır.
XRP (Ripple)	Çok Düşük	Ripple Labs tarafından geliştirilen ve büyük oranda kontrol edilen bir ağıdır. Onaylayıcıların çoğu Ripple ile bağlantılıdır.
EOS	Çok Düşük	Delegated Proof of Stake (DPoS) kullanır. Sadece 21 blok üreticisi vardır, bu da yüksek merkezileşme riski taşır.
HyperLedger Fabric	Yok	Kurumlara özel, izinli bir blokzincirdir. Katılımcılar önceden belirlenmiştir ve tüm ağ süreçleri merkezi şekilde kontrol edilir.

2.3.2. Dağıtık Defter Teknolojisi (DLT)

Geleneksel veri depolama sistemlerinde veriler genellikle merkezi bir veri tabanında toplanmakta ve bu veri tabanının bir veya birkaç yedeği farklı fiziksel lokasyonlarda tutulmaktadır. Bu yöntem, doğal afetler veya siber saldırılar gibi risklere karşı belirli ölçüde koruma sağlamakla birlikte, sistemin temelinde merkezi bir otoriteye ihtiyaç duyulmaktadır. Söz konusu otorite, veri tabanına erişim sağlayarak mevcut verileri değiştirebilmekte,

silebilmekte ya da yeni veriler ekleyebilmektedir. Bu durum, veri bütünlüğü ve güvenliği açısından çeşitli riskler doğurmaktadır.

Dağıtık Defter Teknolojisi (Distributed Ledger Technology - DLT), bu merkezi otoriteye duyulan ihtiyacı ortadan kaldırarak, verilerin güvenliğini ve değiştirilemezliğini dağıtık bir yapı üzerinden sağlamayı amaçlamaktadır. DLT mimarisinin blok zinciri uygulamasında, verilerin bir kopyası ağ üzerindeki tüm düğümler (node) tarafından tutulmaktadır. Bu yapı sayesinde bir verinin değiştirilmesi veya silinmesi neredeyse imkânsız hale gelir; çünkü herhangi bir değişikliğin tüm ağ tarafından doğrulanması gerekmektedir. Ayrıca verinin tamamen ortadan kaldırılabilmesi için ağdaki tüm düğümlerin eş zamanlı olarak yok edilmesi gerekir ki bu da pratikte uygulanması mümkün olmayan bir durumdur. Bu özellikler, DLT'yi yüksek derecede güvenli, şeffaf ve manipülasyona kapalı bir yapı haline getirmektedir.

DLT sürecine gelen teknolojik gelişmelerin nereden başladığına bakmanın konunun anlaşılması için faydalı olacağı kanaatindeyiz. Burada eşler arası iletişim (p2p: peer-to-peer) yani katılımcıların merkezi bir sunucuya bağlı olmaksızın doğrudan birbirleriyle veri paylaşabildiği sistemler mimarisi konunun temelini oluşturmaktadır. Gerçek anlamda kitlesel P2P kullanımı 1999 yılında kullanıcıların kendi bilgisayarlarında bulunan MP3 dosyalarını doğrudan birbirleriyle paylaşmasını sağlayan Napster programı ile başlamıştır. Napster'ın kendi sunucuları olması sebebiyle tam olarak merkeziyetsiz bir yapı olarak nitelendirilmemiştir. Napster sonrası gelişmiş P2P ağları olarak bilinen Gnutella, Kazaa, eDonkey ve BitTorrent gibi uygulamalar bu konuda öncü olmuştur. 2001 yılında kullanılmaya başlanan BitTorrent dosyaları küçük parçalara bölerek çok sayıda kaynaktan aynı anda indirme imkânı sunmuştur. Parça parça veri yayılımı, bugünkü blok zinciri mantığının temel ilkelerinden biridir.

2008 yılında Satoshi Nakamoto tarafından yayımlanan Bitcoin makalesi, P2P ağ kavramını finansal işlemler için kullanarak devrim niteliğinde bir adım atmıştır. Ancak sadece dosya paylaşmakla kalmamış, aynı zamanda verinin doğruluğunu tüm ağ tarafından uzlaşmayla (konsensüs) onaylayan bir sistem sunmuştur. Bu nedenle Bitcoin, klasik P2P sistemlerinden farklı olarak “dağıtık defter teknolojisi (DLT)” mantığına dayalıdır. Yani yalnızca veri

paylaşımı değil, veri kayıtlarının güvenli, değiştirilemez ve merkeziyetsiz biçimde saklanmasını da sağlamaktadır.

Bitcoin, bu teknolojiyi en erken ve etkili kullanan örneklerden biridir. Bitcoin ağında, işlemler blok adı verilen veri yapılarında toplanmakta ve bu bloklar SHA-256 hash fonksiyonu ile birbirine zincirleme şekilde bağlanmaktadır. Yeni bir blok eklenmesi için ağda yer alan madenciler, belirli bir zorluk seviyesini karşılayan hash değerini bulmak zorundadır. Bu süreçte İş Kanıtı (PoW) adı verilir. Hash değeri bulunduktan sonra blok, ağın geri kalanı tarafından doğrulanmakta ve tüm defterlere eklenmektedir. Bu süreç, merkezi bir otoriteye ihtiyaç duymadan güvenli bir kayıt mekanizması oluşturulmasını sağlamaktadır.

Ethereum da benzer bir şekilde çalışmakla birlikte, Bitcoin'den farklı olarak sadece para transferi işlemlerini değil, aynı zamanda akıllı sözleşmeler (smart contracts) adı verilen çalıştırılabilir kod bloklarını da desteklemektedir. Ethereum başlangıçta iş kanıtı (PoW) algoritmasını kullanmaktaydı; ancak zamanla Hisse Kanıtı (PoS) mekanizmasına geçiş yapmıştır. Bu yeni yapıda, doğrulayıcılar (validators) belirli miktarda ETH'yi kilitleyerek ağın güvenliğini sağlamak ve blok üretme hakkı elde etmektedirler. Ethereum da yaşanan bu geçiş, enerji verimliliği ve ağ ölçeklenebilirliği açısından önemli bir adım olmuştur.

DLT ve blok zinciri kavramları çoğu zaman birbirinin yerine kullanılsada, bu iki kavram özdeş değildir. Her blok zinciri bir dağıtık defter teknolojisi örneğidir; ancak her DLT sisteminin blok zinciri olması gerekmemektedir. Örneğin, IOTA gibi bazı sistemler bloklar yerine yönlendirilmiş çevrimsiz grafikler (Directed Acyclic Graph- DAG) kullanarak DLT altyapısını oluşturur. Ayrıca Hyperledger Fabric gibi izinli (permissioned) DLT sistemleri, verilerin yalnızca yetkilendirilmiş katılımcılar tarafından okunabildiği ve yazılabildiği kurumsal çözümler sunmaktadırlar. Bu yapılar blok zinciri mantığını farklı şekilde uygulamakta ya da bazı durumlarda zincir mantığını tamamen terk etmektedirler.

Sonuç olarak, Dağıtık Defter Teknolojisi, güvenli ve merkeziyetsiz veri kaydı imkânı sunarak blok zinciri gibi teknolojilere temel teşkil etmektedir. Blok zinciri ise bu temel üzerinde inşa edilen ve çeşitli uygulama senaryoları için özelleştirilebilen bir altyapı çözümü

olmaktadır. Blok zincirinde kullanılan farklı dağıtık defter yaklaşımları aşağıdaki tabloda verilmiştir.

Tablo 3 : Blok zincirinde farklı dağıtık defter yapıları

Kripto Para	DLT Yapısı	Konsensüs Mekanizması
Bitcoin	Blockchain	Proof of Work
Monero	Blockchain	Proof of Work
Uniswap	Blockchain	Ethereum üzeri, Ethereum'un konsensüsünü kullanır (Proof of Stake)
Maker	Blockchain	Ethereum üzeri, Ethereum'un konsensüsünü kullanır (Proof of Stake)
Ethereum	Blockchain	Proof of Stake
Polkadot	Relay Chain + Parachains	Nominated Proof of Stake (NPoS)
Avalanche	Blockchain (Subnet destekli)	Avalanche Konsensüsü + Snowman
Cardano	Blockchain	Ouroboros (Proof of Stake)
Chainlink	Blockchain Oracle Ağı	Ethereum'un konsensüsü + Off-chain konsensüs
Cosmos	Heterojen Zincirler	Tendermint (Byzantine Fault Tolerance)
Filecoin	Blockchain	Proof of Replication + Proof of Spacetime
Zcash	Blockchain	Proof of Work
Tezos	Blockchain	Liquid Proof of Stake (LPoS)
Solana	Blockchain	Proof of History + Proof of Stake
Optimism	Layer 2 (Rollup)	Ethereum'un konsensüsü + Fraud Proofs
Arbitrum	Layer 2 (Rollup)	Ethereum'un konsensüsü + Multi-round Fraud Proofs
XRP (Ripple)	Ledger (izinli)	Unique Node List (Ripple Protocol Consensus Algorithm)
EOS	Blockchain	Delegated Proof of Stake (DPoS)
Hyper Ledger Fabric	İzinli Ledger	Practical Byzantine Fault Tolerance (PBFT) benzeri

2.3.3. Kriptografi

Kriptoloji, hem şifreleme, şifre çözme ve digital güvenlik araçlarını tasarlayan (kriptografi) hem de tasarlanan bu araçların güvenliklerini test eden (kriptoanaliz) süreçleri kapsayan bir bilim dalıdır. Kriptografi; bilgi güvenliğini sağlamak amacıyla belirli algoritmalar ve anahtarlar kullanarak verilerin şifrlenmesini sağlarken, diğer taraftan kriptoanaliz, kullanılan algoritma ya da anahtar bilgisi olmadan şifreli verilerin çözümüne yönelik yöntemler geliştirmektedir. Tarihsel süreçte insanlar, bilgi alışverişlerinde gizliliği sağlamak amacıyla çeşitli yöntemlere başvurmuşlardır. Bu yöntemlerin zamanla daha karmaşık hâle gelmesinde, şifreli verileri çözmeye çalışan kişilerin sürekli yeni teknikler geliştirmesi etkili

olmuştur. Bilgisayar teknolojilerinin gelişimiyle birlikte hem şifreleme hem de şifre çözme işlemleri daha güçlü ve karmaşık algoritmalar ve sistematik yapılar aracılığıyla gerçekleştirilmeye başlanmıştır.

Blok zinciri sistemlerinde kriptografi, blokların birbirine güvenli biçimde bağlanması, verilerin bütünlüğünün korunması, işlemlerin doğruluğunun sağlanması ve merkeziyetsiz yapının sürdürülebilmesi için vazgeçilmez bir unsurdur. Bu bağlamda veri bütünlüğü çoğunlukla kriptografik özetleme (hash) algoritmaları ile sağlanırken, kimlik doğrulama ve işlemlerin yetkilendirilmesi dijital imzalar aracılığıyla gerçekleştirilir. Gizlilik, şifreleme tekniklerinin yanı sıra sıfır bilgi ispatları (Zero-Knowledge Proofs, ZKP) ile temin edilirken; işlemlerin güvenliği ise asimetrik kriptografi temelli anahtar çiftleri ile sağlanmaktadır.

Blok zincirleri her bloğun, bir önceki bloğun hash (özet) değerini içerecek şekilde oluşturulmasını sağlayarak geçmiş verilerin değiştirilemezliğini garanti altına almaktadır. Bitcoin, hash işlemleri için SHA-256 (Secure Hash Algorithm) algoritmasını kullanırken; Ethereum ağı Keccak-256 algoritmasını kullanmaktadır. Bazı diğer blok zinciri platformlarında ise Blake2b ve SHA-3 gibi alternatif özetleme algoritmaları tercih edilmiştir. Bitcoin ağında ayrıca, cüzdan adreslerinin oluşturulması sürecinde RIPEMD-160 isimli bir başka hash fonksiyonu da kullanılmaktadır.

Kimlik doğrulama ve dijital imzalama süreçlerinde ise asimetrik kriptografi araçları kullanılmaktadır. Blok zinciri sistemlerinde düşük depolama ve işlem maliyeti gerektirdiğinden yaygın olarak Eliptik Eğri Dijital İmza Algoritması (Elliptic Curve Digital Signature Algorithm - ECDSA) tercih edilir. Bu yapı özelinde, özel anahtardan açık anahtar üretimini, eliptik eğriler üzerinde tanımlı matematiksel işlemler aracılığıyla gerçekleştirir. Bu anahtar çifti sayesinde, işlemler dijital olarak imzalanabilir ve kolaylıkla doğrulanabilir. ECDSA'ya alternatif olarak EdDSA (Edwards-curve Digital Signature Algorithm) algoritması da bazı platformlarda kullanılmakta olup, Ed25519 ve Ed448 gibi varyantları bulunmaktadır. RSA algoritması ise, blok zinciri teknolojilerinde yüksek işlem maliyeti nedeniyle nadiren tercih edilmektedir.

Kriptografinin son yıllarda dikkat çeken uygulamalarından biri olan sıfır bilgi ispatları (Zero-Knowledge Proofs, ZKP), kullanıcıların herhangi bir kişisel veri ya da işlem detayını açıklamadan diğer taraflar tarafından doğrulaması işlemi gerçekleştirilmesine olanak tanır. Bu yöntem sayesinde işlem yapan taraf, yalnızca işlem doğruluğunu ispatlayarak gizliliğini koruyabilir. ZKP'nin farklı teknik yaklaşımları bulunmaktadır. Örneğin zk-SNARKs (Zero-Knowledge Succinct Non-Interactive Argument of Knowledge) yöntemi, Zcash gibi mahremiyet odaklı kripto para projelerinde kullanılmakta ve kullanıcıların işlemlerini gizli tutarak onaylamalarını sağlamaktadır. Ethereum ekosisteminde ise ZKP tabanlı ölçeklenebilirlik çözümleri, özellikle Layer-2 protokolleri kapsamında karşımıza çıkmaktadır. ZK-Rollups teknolojisi, ana ağın işlem yükünü azaltarak daha düşük maliyetli ve hızlı işlemler sunmakta, aynı zamanda kullanıcı gizliliğini artırmaktadır. Bu yapıda birçok işlem zincir dışında toplanarak tek bir ispatla ana zincire gönderilmekte, böylece işlem doğruluğu korunurken detaylar açıklanmamaktadır. Sıfır bilgi ispatlarının yaygınlaşmasının en önemli nedeni ise yine blok zinciri teknolojisidir. Çünkü pratikte mahremiyete ihtiyaç duyulan ihtiyaçları blok zincirine kriptografinin bu protokolleri yardımıyla adapte etmek gerekmektedir.

Konsensüs (uzlaşım) mekanizmaları açısından bakıldığında da kriptografik hash işlemleri kritik rol oynamaktadır. Bitcoin ağı, iş kanıtı (Proof of Work- PoW) algoritmasını kullanarak madencilerin SHA-256 fonksiyonu ile zorluk seviyesi belirlenmiş hash problemlerini çözmelerini gerektirir. Ethereum ise, iş kanıtı sisteminden hisse kanıtı (Proof of Stake- PoS) modeline geçerek Casper protokolünü benimsemiştir. Bu protokol, katılımcıların sahip oldukları varlık miktarına ve dijital imzalarına dayalı olarak yeni blokların doğrulanmasını ve ağ güvenliğinin sağlanmasını amaçlamaktadır.

Sonuç olarak, kriptoloji hem teorik hem de uygulamalı boyutlarıyla blok zinciri teknolojilerinin güvenlik, gizlilik ve sürdürülebilirlik ilkelerini mümkün kılan temel bileşenlerden biridir. Kriptografik algoritmaların gelişimi, blok zinciri uygulamalarının daha güvenli, şeffaf ve merkeziyetsiz yapılar üzerine kurulmasına olanak tanımaktadır. Ayrıca kriptoanaliz teknikleri sistemlerin açıklarını kapatmakta ve yaşanabilecek siber saldırıları ihtimalini düşürmektedir. Bölüm 3'te kriptografik temeller ve bu temellerin blok zinciri ilişkileri detaylıca ele alınmıştır.

2.3.4. Şeffaflık ve İzlenebilirlik

Blok zinciri teknolojisi, özellikle şeffaflık ve izlenebilirlik konularında mevcut geleneksel sistemlere kıyasla oldukça iddialı bir alternatif olarak ortaya çıkmaktadır. Dağıtık defter yapısı sayesinde blok zincirine kaydedilen tüm işlemler, internet üzerindeki her katılımcı tarafından görülebilmekte ve doğrulanabilmektedir.

Bitcoin ağı örneği üzerinden değerlendirildiğinde, sistemin ilk bloğundan itibaren gerçekleştirilen tüm işlemler, blok zincirine kronolojik sırayla ve değiştirilemez biçimde kaydedilmektedir. Her bir blok, içerdiği işlemlerle birlikte zaman damgası (timestamp) ile zincire eklenmekte ve bu sayede geçmişe yönelik herhangi bir değişiklik yapılması kriptografik olarak imkânsız hâle gelmektedir. Zincirde yer alan her veri, kriptografik özetleme (hash) fonksiyonları aracılığıyla bir önceki blokla ilişkilendirildiğinden, en küçük bir değişiklik bile zincirin bütünlüğünü bozar ve bu durum ağdaki diğer düğümler tarafından anında fark edilir. Bu yapı yalnızca sistemin güvenliğini değil, aynı zamanda işlem geçmişinin doğruluğunu ve izlenebilirliğini de güvence altına almaktadır.

Blok zinciri ağlarının açık ve izin gerektirmeyen yapısı sayesinde, herhangi bir kullanıcı ek bir onaya ihtiyaç duymaksızın ağa katılabilmekte, işlemleri görüntüleyebilmekte ve doğrulama süreçlerine katkıda bulunabilmektedir. Bu durum, klasik merkezi sistemlerin aksine kullanıcıya tam şeffaflık sağlamaktadır. Örneğin, geleneksel bankacılık sistemlerinde kullanıcılar yalnızca kendi işlemlerine sınırlı olarak ulaşabilirken, blok zincirinde sistemin tüm geçmişine erişim mümkündür.

Gerçek zamanlı olarak tüm işlemlerin izlenebilmesi, özellikle finansal işlemler, tedarik zincirleri ve kamu yönetimi gibi alanlarda yolsuzluk ve usulsüzlük riskini önemli ölçüde azaltma potansiyeline sahiptir. Gelişmiş sorgulama araçları ve blok gezginleri (block explorer) sayesinde hem bireyler hem de bağımsız denetçiler, ağdaki tüm işlem geçmişine ulaşabilmekte ve bu verileri analiz edebilmektedir.

Ancak burada göz ardı edilmemesi gereken önemli bir nokta, şeffaflığın bir avantaj olmakla birlikte bazı durumlarda bireysel gizlilikle (mahremiyetle) çelişebileceğidir. Bu nedenle Ethereum gibi bazı ağlarda, gizlilik artırıcı protokoller ve ZKP (Zero-Knowledge Proofs) gibi teknolojiler ile gerekli durumlarda gizlilik ihtiyacı karşılanmaktadır. Böylece işlemler doğrulanabilmekte ancak içerikleri gizli tutulabilmektedir.

2.3.5. Değiştirilemezlik (Immutability)

Blok zinciri teknolojisinin sunduğu en kritik güvenlik ilkelerinden biri, sistemde tutulan verilerin değiştirilemez (immutability) olmasıdır. Bu özellik, merkezi bir otoriteye ihtiyaç duymadan verilerin güvenliğini sağlama kabiliyeti sayesinde, blok zinciri sistemlerini mevcut geleneksel veri yapılarından ayıran temel farklardan biri olmaktadır. Özellikle merkeziyetsiz yapılarda veri güvenliğinin, sistemin bütünlüğünü ve güvenilirliğini sürdürebilmek için hayati öneme sahip olduğu açıktır.

Blok zincirinde değiştirilemezlik, her bloğun kendisinden önceki bloğun kriptografik özetini (hash değerini) içermesi sayesinde sağlanırken blokların birbirine zincir halkaları gibi bağlanmasına neden olmaktadır. Herhangi bir blok üzerinde yapılan en küçük bir değişiklik, o bloğun hash değerini, dolayısıyla sonraki tüm blokların hash dizisini geçersiz hale getirmektedir. Bu değişiklik, ağdaki tüm düğümler (node) tarafından hemen fark edilerek reddedilmektedir. Böylece blok zincirinde bir bloğa bir kez yazılan veri, pratik olarak geri döndürülemez ve değiştirilemez hale gelmektedir.

Verilerin değiştirilemezliği yalnızca teknik değil; aynı zamanda hukuki ve toplumsal açıdan da önemli sonuçlar doğurmaktadır. Örneğin, bir işlemin blok zincirine kaydedilmesi, o işlemin zaman damgası ile birlikte şeffaf ve kalıcı biçimde kayıt altına alınmasını sağlamaktadır. Bu durum; mülkiyet kayıtları, sözleşmeler, oy verme sistemleri ve finansal işlemler gibi pek çok alanda güvenin tesis edilmesine zemin hazırlamaktadır.

Blok zincirinin dağıtık yapısı da değiştirilemezlik ilkesinin uygulanmasında önemli bir rol oynamaktadır. Veriler, tek bir merkezi sunucuda değil; ağı oluşturan tüm düğümler üzerinde saklanmaktadır. Bu durum, sistemin tek bir noktadan saldırıya uğramasını engellediği gibi,

kötü niyetli bir kullanıcının geçmişi değiştirme girişimini son derece maliyetli ve neredeyse imkânsız hale getirmektedir. Özellikle Bitcoin gibi konsensüs algoritması olarak iş kanıtı (Proof of Work) kullanan sistemlerde, geçmiş bir bloğu değiştirebilmek için o bloğun ve ondan sonraki tüm blokların yeniden işlenmesi (mining) gerekmektedir. Bu durumda devasa hesaplama gücü ve enerji sarfiyatı gerekmektedir. Ancak bu işlem ekonomik olarak rasyonel bir hareket olmadığından, gerçekleştirilebilirliği mümkün bulunmamaktadır.

Bununla birlikte, şeffaflığın bireysel gizlilikle çelişmesi gibi, değiştirilemezlik de bazı durumlarda esneklik gereksinimi ile çelişebilmektedir. Özellikle hatalı veri girişi, hukuki düzenlemeler (örneğin “unutulma hakkı”) veya dolandırıcılık gibi durumlarda, verilerin değiştirilemez olması bir sorun haline gelebilmektedir. Bu gibi durumlar için bazı blok zinciri projeleri, "ekleyerek düzeltme" (append-only correction) veya "soft fork" gibi yöntemlerle sistemi manipüle etmeden düzeltme imkânı sağlamaya çalışmaktadır. Ancak genel prensip, bir verinin bir kez kaydedildikten sonra doğrudan silinemez ya da değiştirilemez olmasıdır.

Değiştirilemezlik ilkesi, blok zincirine duyulan güvenin temelini oluşturmaktadır. Bu ilke sayesinde kullanıcılar, geçmişte gerçekleştirdikleri işlemlerin kayıtlı kalacağını bilmektedirler. Bu da özellikle finans, hukuk ve kamu yönetimi gibi alanlarda blok zinciri teknolojisinin daha fazla benimsenmesini teşvik etmektedir.

2.3.6. Konsensüs Mekanizmaları

Blok zinciri teknolojisinde konsensüs mekanizmaları karar verme süreçleri için kullanılan mekanizmalardır. Düğümler oluşturulacak yeni blokların doğruluğunu onaylamak, çift harcamayı önlemek ve ağın güvenliğini sağlanması için bu mekanizmaları kullanırlar.

Bu mekanizmalardan en çok kullanılanlar şunlardır;

İş Kanıtı (PoW- Proof of Work): Bu mekanizmada madenciler matematiksel bir problemi çözmeye çalışırlar. Hesaplamaya dayalı olduğu için bu sistemlere saldırılar oldukça

maliyetlidir. Merkezi bir otoriteye ihtiyaç duyulmamaktadır. En bilinen örneği Bitcoin ağıdır. Enerji tüketimi ve ölçeklenebilirlik konusunda sıkıntılar yaşanabilmektedir.

Hisse Kanıtı (PoS- Proof of Stake): Bu mekanizmada düğümler sahip oldukları kripto para miktarı oranında blok üretme hakkına sahip olurlar. Yüksek hesaplama gücü gerektirmediği için enerji verimliliği ve ölçeklenebilirlik konusunda daha başarılıdır. Daha fazla kripto paraya sahip olan düğümlerin daha fazla hakka sahip olması merkezileşme riskini artırmaktadır. En bilinen örneği Ethereum ağıdır. Ethereum PoW dan PoS’a geçmiştir.

Temsilcili Hisse Kanıtı (DPoS- Delegated Proof of Stake): Bu mekanizmada belli sayıda temsilci düğüm üzerinden blok üretimi yapılır. Ağ katılımcıları güven duydukları düğümleri temsilci olarak seçer. Seçilen temsilciler sırayla blok üretir ve işlemleri doğrular. Enerji tüketimi düşüktür ve sınırlı sayıda temsilci nedeniyle daha hızlı blok üretilir. Ancak az sayıda temsilci merkeziyetçilik riski doğurmaktadır. En bilinen örnekler EOS ve TRON ağlarıdır.

Tarih Kanıtı (PoH- Proof of History): Blok zinciri ağında zaman damgası ve işlem sırasını garanti altına almak için kullanılan bir konsensüs mekanizmasıdır. PoH, Solana blok zinciri tarafından geliştirilmiştir ve zamanın kendisini bir veritabanına kaydederek bloklar arasındaki işlem sırasını kesin olarak doğrular. Bu mekanizma genellikle diğer konsensüs mekanizmalarıyla birlikte çalışır (örneğin, Solana’da PoH, Hisse Kanıtı ile entegre bir şekilde kullanılır).

Yetki Kanıtı (PoA- Proof of Authority): İşlemleri doğrulayan düğümlerin kimliklerinin güvenilir kişi ya da kurumlar tarafından önceden onaylandığı bir konsensüs mekanizmasıdır. Bu modelde düğüm sayısı azdır ve blok üreticileri belirli bir “yetki”ye sahiptir. Enerji tüketimi düşüktür ve yüksek işlem hızı sağlar. Merkeziyetlilik riski çok yüksektir. Ronin Network, POA network, BSC (Binance Smart Chain teknik olarak PoA ve DPoS karışımı kullanır, az sayıda doğrulayıcıyla çalışır.) bu mekanizmayı kullanan blok zinciri ağlarına örnek olarak verilebilir.

Önem Kanıtı (PoI – Proof of Importance): Proof of Importance, yalnızca token miktarına değil, bir kullanıcının ağ içindeki etkileşimi, işlem hacmi ve katkı düzeyi gibi sosyal ve ekonomik aktivitesine göre blok üretme hakkı tanıyan mekanizmadır. Bu, klasik PoS'tan farklı olarak daha adil ve katkı bazlı bir yapı sunar. Pasif coin sahiplerini değil, aktif kullanıcıları ödüllendirirken merkeziyetsizliği ve katılımı artırır. NEM ve SYMBOL zincirleri bu konsensüs mekanizmasını kullanmaktadır.

2.3.7. Cüzdanlar (Wallet)

Blok zinciri teknolojisinin temel yapı taşlarından biri olan kripto cüzdanlar, kullanıcıların dijital varlıklarını (örneğin kripto paralar, tokenler, NFT'ler vb.) yönetmelerini sağlayan yazılım ya da donanım tabanlı araçlardır. Ancak kamuoyunda yaygın olan yanlış kanının aksine, bu cüzdanlar kullanıcıların varlıklarını fiziksel olarak tutmamaktadırlar. Kripto varlıklar, her zaman kendi blok zincirleri üzerinde kayıtlı olarak durmakta; cüzdanlarda ise yalnızca bu varlıkların üzerinde hak sahibi olmayı sağlayan özel anahtarlar (private key) saklanmaktadır.

Bu bağlamda, bir cüzdan yalnızca belirli bir adresteki varlıklara erişim yetkisi tanımaktadır. Örneğin, bir Bitcoin'i bir adresten alıp başka bir adrese fiziksel anlamda "taşımak" teknik olarak mümkün olmamaktadır. Gerçekte yapılan, yalnızca o varlığı kullanmaya yetkili olan özel anahtarın (ve bu anahtara bağlı hesap yapısının) değiştirilmesidir. Bu nedenle, cüzdanlar blok zinciri ekosisteminde oldukça karmaşık olan hesap, adres ve anahtar yönetimini kullanıcı adına gerçekleştiren araçlar olarak işlev görmektedir.

Bir kripto cüzdan, temelde şu üç bileşeni yönetmektedir:

- **Özel Anahtar (Private Key):** Kriptografik olarak işlem imzalamaya yarayan ve gizli tutulması gereken bilgidir. Cüzdandaki dijital varlıkları harcama yetkisi bu anahtarla sağlanmaktadır.
- **Açık Anahtar (Public Key):** Özel anahtardan türetilen, herkesle paylaşılabilen kriptografik bilgidir. Genellikle hash'lenerek cüzdan adresi oluşturulmaktadır.
- **Cüzdan Adresi:** Kullanıcının blok zinciri üzerindeki tanımlayıcı kimliğidir. Public key'in hash'lenmesiyle elde edilmekte ve dış dünyayla etkileşimde kullanılmaktadır.

Bir cüzdan oluşturulurken, öncelikle rastgele bir özel anahtar üretilmektedir. Bu anahtardan açık anahtar türetilip; açık anahtar bir hash fonksiyonundan geçirilerek cüzdan adresi oluşturulmaktadır. Cüzdan yazılımı bu bilgileri şifreli biçimde saklamakta ve kullanıcının işlem gerçekleştirebilmesi için bu anahtarı kullanarak imza üretimini sağlamaktadır. Blok zincirine yazılan her işlem bu imza sayesinde doğrulanarak, böylece işlem güvenliği sağlanmış olmaktadır.

Kripto cüzdanlar genel olarak bağlantı tiplerine göre iki ana gruba ayrılmaktadır:

- **Sıcak Cüzdanlar (Hot Wallets):** İnternete bağlı olan cüzdanlardır. Kullanım kolaylığı sağlasa da, sürekli çevrimiçi olmaları nedeniyle siber saldırılara karşı daha savunmasızdırlar. Örnekler arasında Metamask, Trust Wallet, tarayıcı uzantıları ve masaüstü yazılımlar yer almaktadır.
- **Soğuk Cüzdanlar (Cold Wallets):** İnternete doğrudan bağlı olmayan cüzdanlardır. Fiziksel güvenlik ön plandadır ve uzun vadeli saklama için tercih edilmektedirler. İki ana alt sınıfa ayrılmaktadır:
 - **Donanım Cüzdanları:** Fiziksel cihazlardır (örneğin Ledger, Trezor).
 - **Kâğıt Cüzdanlar:** Private key'lerin QR kod veya düz yazı biçiminde kâğıt üzerinde saklandığı yöntemdir.

Cüzdan güvenliği açısından kritik bir unsur da Seed Phrase (kurtarma cümlesi) kavramıdır. Bu cümle genellikle BIP-39 standardına göre seçilen 12 ya da 24 adet İngilizce kelimeden oluşur ve özel anahtarın yeniden oluşturulmasına imkân tanımaktadır. Mnemonic yapılar, özel anahtarları kullanıcıların daha kolay saklayabilmesi ve yedekleyebilmesi için geliştirilmiştir. Seed phrase, çevrimdışı bir ortamda güvenli bir şekilde saklanmalıdır. Bu bilginin kaybolması durumunda cüzdan ve içindeki tüm varlıklar erişilemez hâle gelmektedir.

Modern cüzdanlar, yalnızca dijital varlıkları saklamakla kalmayıp, aynı zamanda akıllı sözleşmelerle de etkileşime geçebilmektedir. Özellikle Metamask, bu anlamda yaygın

şekilde kullanılan bir tarayıcı uzantısıdır. Kullanıcı, Metamask üzerinden bir merkeziyetsiz uygulama (DApp) ile işlem yapmak istediğinde, işlem ayrıntıları cüzdan tarafından kullanıcıya sunulmakta ve işlem, özel anahtarla imzalanarak blok zincirine yazılmaktadır. Bu etkileşimlerde genellikle Web3.js veya Ethers.js gibi kütüphaneler kullanılmaktadır.

Blok zinciri teknolojisinin gelişmesiyle birlikte yalnızca Bitcoin ve Ethereum gibi ana kripto paralar değil, aynı zamanda çok sayıda token (ERC-20, BEP-20), NFT (ERC-721, ERC-1155) ve farklı zincirlerdeki dijital varlıklar da hayatımıza girmiştir. Her biri için ayrı cüzdanlar kullanmak kullanıcılar açısından karmaşık ve yönetimi zor hale gelmiştir. Bu nedenle, çoklu varlık desteği sunan modern cüzdan yazılımları geliştirilmiştir. Metamask bu alanda en bilinen örneklerden biridir ve Ethereum başta olmak üzere tüm EVM uyumlu zincirlerle çalışabilmektedir.

Son dönemde cüzdan altyapılarında önemli teknolojik gelişmeler yaşanmaktadır:

- **Smart Wallets (Akıllı Cüzdanlar):** Cüzdanın kendisinin bir akıllı sözleşme olarak tanımlandığı yapılardır. Örneğin Account Abstraction destekleyen cüzdanlar (Argent, Safe) bu kapsamdadır. Bu sayede işlem imzalama, harcama limitleri ve sosyal kurtarma (social recovery) gibi gelişmiş özellikler desteklenebilir.
- **Multi-Signature Wallets (Çok İmzalı Cüzdanlar):** Bir işlemin gerçekleşmesi için birden fazla imza gerektiren sistemlerdir. Kurumsal kullanım için oldukça uygundur.
- **MPC (Multi-Party Computation) Cüzdanlar:** Özel anahtarın birden fazla parçaya bölünerek farklı kişiler ya da cihazlarda saklandığı yapılardır. Bu modelde anahtarın tamamı tek bir kişi tarafından bilinmemekte; dolayısıyla kötü niyetli bir aktör tek başına işlem yapamamaktadır.

Blok zinciri teknolojisinin temel ilkelerinden biri, üçüncü taraflara güven ihtiyacını ortadan kaldırmaktır. Ancak, ortalama bir kullanıcının anahtar yönetimi, imzalama ve akıllı sözleşme etkileşimleri gibi işlemleri teknik bilgi gerektirdiğinden, birçok kullanıcı pratikte cüzdan üreticilerine güvenmek zorunda kalmaktadır. Bu durum, teknolojinin teorik hedefleri ile pratik uygulamaları arasında bir çelişki yaratmaktadır.

2.3.8. Borsalar

Blok zinciri teknolojisinin temel bileşenlerinden biri olan borsalar, bir varlığın fiyatının belirlenmesinde merkezi rol oynayan pazar yerleridir. Blok zinciri teknolojisinin en yaygın uygulamalarından biri olan kripto paraların fiyatlarının oluşumu ve geniş kullanıcı kitlelerine ulaşması, bu platformlar sayesinde mümkün hale gelmiştir. Bu açıdan bakıldığında, kripto para borsalarının rolü sadece ticari değil, aynı zamanda teknolojinin geniş kitlelerce benimsenmesi açısından da stratejik bir öneme sahiptir.

Kripto para borsalarının temel faydalarını şu şekilde sıralayabiliriz:

- Kripto para fiyatlarının piyasa şartlarında oluşmasını sağlayarak, piyasa derinliğinin artmasıyla fiyat stabilizasyonuna katkıda bulunurlar.
- Teknolojik okuryazarlığı yüksek olmayan kullanıcıların dahi kripto paralara kolayca erişimini sağlarlar. Böylece geniş halk kitleleri finansal işlemler yapabilmektedir.
- Yeni blok zinciri projeleri için finansal kaynak sağlamada aracı rol üstlenirler.
- Merkeziyetsiz finans (DeFi) uygulamalarının hayata geçirilmesi ve yaygınlaştırılmasında temel rol oynarlar.

Blok zinciri tabanlı borsaların henüz olmadığı ilk dönemlerde, özellikle Bitcoin ağının ilk yıllarında, BTC işlemleri bireyler arasında doğrudan gerçekleştirilmekteydi. Bu süreçte kullanıcılar, Bitcointalk gibi forumlarda OTC (Over the Counter) işlemlerle alım-satım gerçekleşmekte ve fiyatlar karşılıklı müzakerelerle belirlenmekteydi. Ancak bu yöntemin pratik zorlukları, yüksek güvenlik riskleri ve fiyat belirleme güçlükleri, kripto paraların geniş kitlelere ulaşmasını sınırlandırmıştır.

2010 yılında ilk resmi Bitcoin borsası olarak kabul edilen BitcoinMarket.com platformu hizmete açılmış ve BTC/USD paritesini listeleyerek ilk kez standartlaştırılmış bir fiyat oluşturmuştur. Ancak bu platform, teknik altyapısındaki eksiklikler, likidite problemleri ve o dönemde Bitcoin'e olan talebin henüz yeterli düzeyde olmaması sebebiyle kısa sürede faaliyetlerini durdurmak zorunda kalmıştır.

Aynı yıl Japonya merkezli olarak açılan Mt. Gox borsası, daha profesyonel yönetim anlayışı ve kullanıcı dostu ara yüzü ile büyük bir popülerite kazanmıştır. Bitcoin ve blok zinciri teknolojisine artan ilgi ile birlikte, Mt. Gox hızla dünyanın en büyük Bitcoin borsasına dönüşmüş ve 2013 yılında tüm Bitcoin işlemlerinin yaklaşık %70'ini gerçekleştirmeye başlamıştır. Ancak 2014 yılında yaşanan büyük çaplı güvenlik ihlali borsanın iflasına neden olmuş ve kripto para borsalarına yönelik ciddi güven sorunlarını gündeme gelmesine sebebiyet vermiştir. Benzer güvenlik zafiyetleri, aynı dönemde faaliyet gösteren Bitstamp ve BTC-e gibi borsaların itibarını da zedelemiş ve faaliyetleri durdurulmuştur. Ayrıca Silk Road gibi yasadışı faaliyetlerin yürütüldüğü platformların varlığı da kamuoyu algısını olumsuz etkileyerek insanların borsalara güvenini azaltmıştır.

Bu kriz ve güvensizlik döneminden sonra, sektörde daha profesyonel ve güçlü altyapılarla donatılmış yeni nesil merkezi borsaların (Centralized Exchange- CEX) kurulması dönemi başlamıştır. Coinbase (2012), Kraken (2011), Binance (2017), OKX (2017) ve KuCoin (2017) gibi günümüzde halen faaliyet gösteren büyük borsalar, kurumsal yapıları ve yüksek güvenlik standartlarıyla kripto paraların daha geniş kitlelere ulaşmasını ve piyasanın derinleşmesini sağlamıştır. Bugün bu borsalar günlük milyarlarca dolarlık işlem hacmine ulaşmış durumdadır.

Bu dönemde, blok zinciri teknolojisinin finansmanı için ICO (Initial Coin Offering) yöntemi yaygınlaşmış ve birçok yeni kripto para projesi, borsalarda listelenerek kullanıcılarla buluşmuştur. ICO'ların ardından süreç, daha kontrollü ve güvenli olan IEO (Initial Exchange Offering) formatına evrilmiştir. IEO modelinde, borsalar projelerin finansmanında aktif bir rol üstlenmiş ve yatırımcı güvenini artırmıştır.

Son yıllarda, merkeziyetsiz borsa (Decentralized Exchange- DEX) kavramının yükselişi dikkat çekmektedir. Merkeziyetsiz borsaların en bilinen örneği olan Uniswap (2020), kullanıcıların doğrudan birbirleriyle merkezi bir otorite olmadan işlem yapabilmelerini v4 sürümüyle 12 blok zinciri üzerinde sağlamaktadır. Ayrıca PancakeSwap (2020- BSC, Ethereum, Aptos), SushiSwap (2020- Ethereum, Arbitrum, Polgon, Avalanche, BSC) ve dYdX (2017 – Ethereum, dYdX Chain) gibi Dex borsaları da günümüzde aktif olarak

kullanılmaktadır. Bu yeni model, finansal özgürlük, şeffaflık ve güvenliği ön plana çıkararak merkezi borsaların karşısında önemli bir alternatif haline gelmiştir.

Son dönemde ise ulusal ve düzenleyici çerçevelere uygun borsalar yaygınlaşmaktadır. Dünyanın farklı bölgelerinde kripto paraların yasal statüsünün netleşmesiyle birlikte, düzenleyici kurumlar (örneğin Türkiye’de Sermaye Piyasası Kurulu, ABD’de SEC, Avrupa Birliği’nde MiCA düzenlemesi) gözetiminde faaliyet gösteren ve lisanslanan borsalar ortaya çıkmıştır. Bu durum, kullanıcı güveninin artmasına ve kripto para piyasasının daha sağlam temeller üzerine oturmasına olanak tanımaktadır.

Sonuç olarak, borsalar, blok zinciri teknolojisinin geniş kitleler tarafından benimsenmesi ve kripto varlıkların daha istikrarlı ve erişilebilir hale gelmesi açısından temel bir rol üstlenmektedir. Kullanıcı talepleri borsaların ortaya çıkışını teşvik ederken, borsaların varlığı da kullanıcıların kripto paralara daha kolay ve güvenli şekilde erişmelerini sağlayarak karşılıklı bir büyüme dinamiği oluşturmaktadır.

Tablo 4: Günlük işlem hacmine göre kripto borsaları

	Borsa	Güven Puanı	24sa Hacim	Aylık Ziyaretçi Sayısı
1	 Binance	10/10	\$17.014.376.568	57,6 M
2	 Coinbase Exchange	10/10	\$3.434.123.379	40,2 M
3	 Bybit	10/10	\$3.341.744.813	18,7 M
4	 Bitget	10/10	\$3.077.959.214	21,8 M
5	 OKX	10/10	\$3.061.470.421	26,1 M
6	 Crypto.com Exchange	10/10	\$2.095.919.119	9,18 M
7	 Kraken	10/10	\$1.171.813.203	6,66 M

25.04.2025 tarihi itibarıyla www.coingecko.com sitesinden alınmıştır.

Kripto borsalar hakkında istatistiksel bilgi veren önemli sitelerden biri olan Coingecko’ya göre 25.04.2025 tarihinde sitenin takip ettiği 216 kripto borsada toplam 96.2 milyar Dolar işlem hacmi gerçekleşmiştir. Yukarıdaki tabloda site kriterlerine göre tam güvenli olarak

sınıflandırılan borsalardan günlük işlem hacmi 1 milyar Doların üzerinde olan borsalar listelenmiştir.

2.3.9. Akıllı Sözleşmeler

Bitcoin ile blok zinciri teknolojisi kendi rüştünü ispatlayarak merkeziyetsiz bir sistemin nasıl çalışabileceğini ortaya koymuş ve bir devrim etkisi yaratmıştır. Bu teknolojinin potansiyeli gören bazı Bitcoin geliştiricilerinin blok zinciri ile daha neler yapılabileceği konusundaki yaklaşımları kısa zaman sonra yeni bir blok zinciri olan Ethereum ile akıllı sözleşmeleri hayatımıza sokmuştur. Böylelikle sadece bir para ya da değer olmanın dışında merkeziyetsiz olarak otomatik çalışabilen sistemler hayatımıza girmiştir.

Akıllı sözleşme, önceden belirlenmiş kurallar ve koşullar çerçevesinde otomatik olarak çalışan, dijital ortamda yazılmış bir bilgisayar programıdır. Taraflar arasındaki anlaşmalar, bu program aracılığıyla otomatik olarak yürütülür ve koşullar sağlandığında sözleşmeye bağlı işlemler (para transferi, veri güncelleme...) otomatik olarak gerçekleştirilir.

Blok zinciri üzerinde oluşturulmanın doğal sonucu olarak akıllı sözleşmeler bazı temel özelliklere otomatik olarak sahiptirler. Öncelikle insan müdahalesi olmadan otomatik olarak çalışırlar. Blok zinciri teknolojisinin sağladığı kriptografik güvenlik sayesinde sözleşme verileri ve işlemler güvence altındadır. Blok zinciri üzerinde yer aldıklarından herkes tarafından görülebilir ve denetlenebilirler. Yine merkeziyetsizlik sayesinde bütün katılımcılar tarafından ulaşılabilir ve kullanılabilirlerdir.

Bütün bu olumlu özelliklerin yanında sözleşmenin oluşturulması sırasında yapılan kodlama hataları, yapılan işlemlerin geri dönülemez olması ve yasalar ve düzenlemelere tabi olmadığı için sözleşmelerin hukuksal bir statüsünün olmaması dikkate alınması gereken dezavantajlar olarak karşımıza çıkmaktadır.

2.3.10. Katmanlar

Katmanlar konusu blok zinciri teknolojisinde oldukça önemli bir konudur. Katman 0 (Layer 0), Katman 1 (Layer 1), Katman 2 (Layer 2) ve Katman 3 (Layer 3) olarak 4 başlıkta incelenebilirler ancak ortaya çıkma zamanları açısından incelenemenin daha anlaşılır olacağı kanaatindeyiz.

Katman 1 ana blok zinciri katmanı olarak ifade edebileceğimiz katmandır. Bitcoin'in ilk ortaya çıkması ile oluşmuştur. Kendi zincirleri ve kendi doğrulama mekanizmaları olan sistemlerdir. En bilinen örnekler Bitcoin ve Ethereum'dur.

Katman 2 ise Katman1 de yaratılan sistemlerin ölçeklenebilme sorununu çözebilmek yani işlem hızını artırmak ve işlem ücretlerini düşürmek üzere geliştirilmiş bir çözümdür. Ana blok zincirin güvenliğinden faydalanarak daha hızlı işlem ve düşük maliyet sağlamaktadır. Bitcoin Lightning network, Ethereum Raiden Network, Optimizm, Arbitrum, zkSync, StarkNet örnek olarak verilebilir.

Katman 3, Katman 1 ve Katman 2 üzerine inşa edilen akıllı sözleşmeler ve kullanıcı arayüzleri içeren katman olarak tanımlanır. DeFi, NFT Pazar yeri ve oyunlar gibi uygulamalar içerir. Uniswap (Ethereum – DeFi), Aave (Ethereum – DeFi), OpenSea (NFT Pazarı), Axie Infinity (Oyun) ve Decentraland (Metaverse) birkaç örnek olarak sayılabilir.

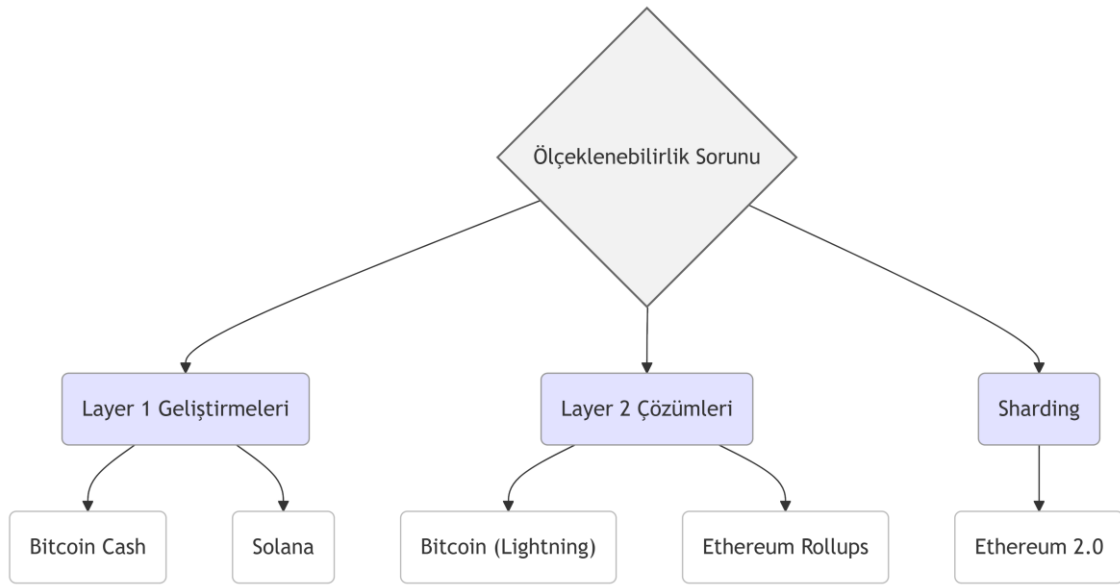
Katman 0 birden fazla Katman 1 blok zinciri oluşturulmasını ve birlikte çalışmasını sağlayan bir altyapı katmanıdır. Palkadot (DOT) ve Cosmos (ATOM) Katman 0 örnekleridir.

2.3.11. Ölçeklenebilme ve trilemma

Ölçeklenebilirlik (scalability), sistemin yüksek sayıda işlemi hızlı ve verimli bir şekilde işleyebilme kapasitesini ifade etmektedir. Sistemler bir süre sonra ilk tasarlandıkları seviyenin üstünde işlem hacmi, kullanıcı sayısı gibi durumlarla karşılaştıklarında problem yaşamaktadırlar. Bitcoinin saniyedeki işlem sayısının ve bir blokta tutabildiği işlem

sığasının yetersiz olması gibi projenin yaygınlaşmasından sonra ortaya çıkan sorunlar buna en güzel örnektir.

Şekil 2: Blok zinciri teknolojilerinde ölçeklenebilme soru



Ölçeklenebilme blok zinciri teknolojisinin en önemli zorluklarından biridir. Bu sorun için üç farklı çözüm yöntemi olduğu söylenebilir. Birincisi, Katman 1 çözümler dediğimiz doğrudan ana zincir protokolünün iyileştirilmesi çalışmalarıdır. Bitcoin Cash örneğinde olduğu gibi blok boyutunun büyütülmesi ve Solana örneğinde olduğu gibi daha hızlı konsensüs mekanizmaları kullanılması bu tip çözümlere örnek olarak gösterilmektedir. İkinci çözüm yaklaşımı ise Katman 2 çözümleridir. Burada ana zincirin üzerine ek bir katman eklenerek ana zincirin iş yükü azaltılmaktadır. Bitcoin’de de Lightning Network ile mikro ödemeler hızlı yapılmaktadır. Bir başka örnek ise Ethereum da Rollups ile işlemlerin topluca işlenip sonra ana zinciri yazılmasıdır. Ölçeklenebilirlik problemlerinde son çözüm yöntemi ise sharding yani ağın parçalanması yaklaşımıdır. Bu yaklaşımla Ethereum 2.0 ile ağ 64 shard’a bölünmüştür.

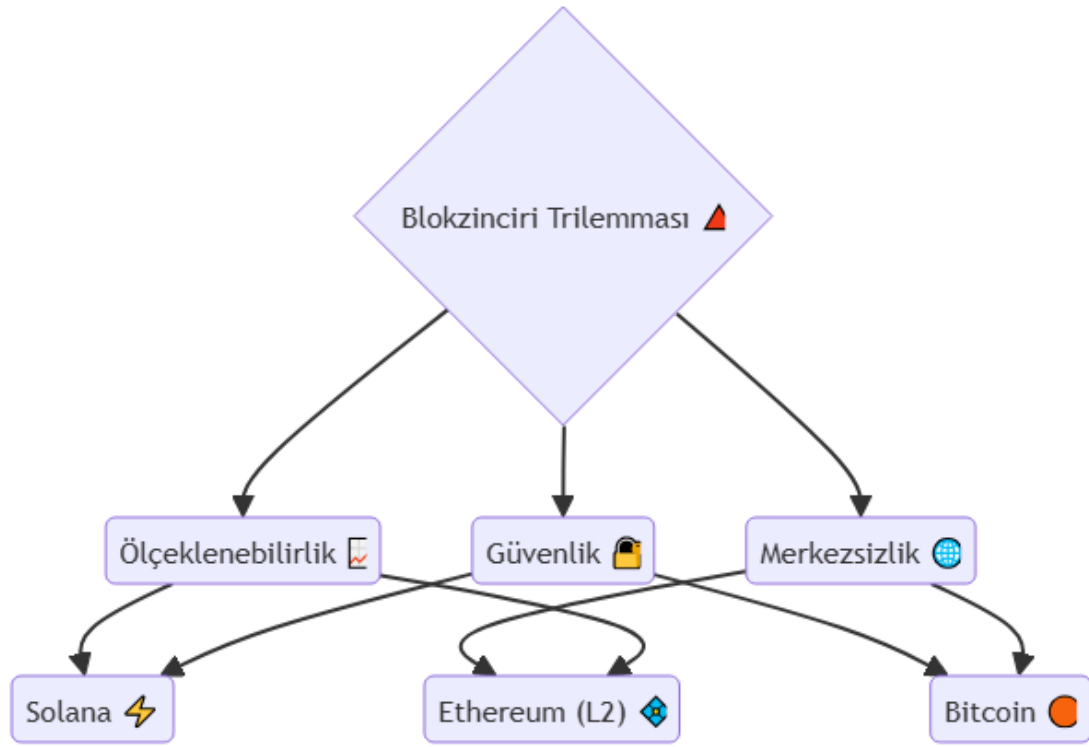
Görüldüğü üzere ölçeklenebilirlik problemi yeni yaklaşımlar ve teknolojilerle her katmanda çözüm üretilmeye çalışılan bir zorluktur. Burada ölçeklenebilme ile güvenlik ve

merkeziyetsizliğin ilişkisi de önem kazanmaktadır. Bu durum iteratürde “blok zinciri trileması” olarak adlandırılan yapısal probleme işaret etmektedir.

Blok zincirinde trilemma kavramı Ethereum’un kurucularından Vitalik Bulterin tarafından ortaya konulmuş bir kavramdır. Vitalik burada bir blok zincirinin en önemli birleşenleri olan merkeziyetsizlik, güvenlik ve ölçeklenebilirlik kavramlarının birbirlerine göre durumları göz önünde bulundurarak bu alanların herhangi birinde yapılan değişikliğin diğer alanları da etkileyeceğini ortaya koymaktadır. Bu üç alanda aynı anda tam olarak mükemmel olamaz. Çünkü birinde yaptığınız geliştirme diğer alan da zafiyetlerin artmasına yol açmaktadır.

Örneğin merkeziyetsizlik ve güvenlik seviyesini artırdıkça ölçeklenebilirlik problemleri ortaya çıkmaktadır. Bu durum Ethereum ve Bitcoin de rahatlıkla gözlemlenebilmektedir. Güvenlik ve ölçeklenebilmeyi üst seviyede kurguladığınızda sistem giderek daha merkeziyetçi bir yapıya dönüşmektedir. Bu durumu görebileceğimiz örnek olarak merkezi node yapısı kullanan BNP (Binance Smart Chain yapısında 21 doğrulayıcı vardır.) incelenebilir. Merkeziyetsizlik ve ölçeklenebilmenin ön planda tutulduğu durumlarda ise güvenlik riskleri oluşmaktadır. Bu açıdan bakıldığında Ethereum’un ölçeklenebilme problemlerini çözen Katman 2 yapılar (Optimism, zkRollup gibi) oluşturdukları ek katmanlarla güvenlik riskini artırmaktadırlar.

Şekil 3: Blok zinciri trilemması



Bu trilemma aslında bir tasarım sınırlamasıdır. Her kullanıcının node çalıştırabildiği, güvenli bir zincir istenirse; işlem kapasitesinden ödün verilecektir. Ölçeklenebilirliği artırmak, daha güçlü donanım gerektirir ve katılım zorlaşır; bu da merkeziyetsizliği zedelemektedir. Sonuç olarak her blok zinciri projesi kendi tasarımında önceliklerini belirlemelidir.

3. BLOK ZİNCİRİ TEKNOLOJİSİNE MATEMATİKSEL BİR BAKIŞ

Blok zinciri teknolojisi, uzun yıllar boyunca gelişim göstermiş olan kriptografi biliminin günümüzdeki en önemli uygulamalarından biri olarak değerlendirilmektedir. Bu teknoloji, temelinde güçlü bir şekilde kriptografik yöntemlere dayanmakta olup, matematiksel güvenlik prensiplerini esas almaktadır.

Bu bölümde, blok zinciri teknolojisinde sıklıkla başvurulanan kriptografik temeller ele alınacaktır. Konular, özellikle blok zinciri uygulamaları bağlamında önem arz eden başlıklar çerçevesinde özetlenecektir.

Blok zinciri sistemlerinin temel yapı taşlarından biri, veri bütünlüğünün eksiksiz biçimde sağlanmasıdır. Bu durum, aynı zamanda kriptografinin de temel amaçlarından biri olarak öne çıkmaktadır. Dolayısıyla, bir sonraki alt bölümde, kriptografi kapsamında veri bütünlüğünü sağlamada yaygın olarak kullanılan özet (hash) fonksiyonları ayrıntılı biçimde ele alınacaktır.

3.1. Özetleme Fonksiyonları

Dijital dünyada birçok veri türü bulunmaktadır ve bu verilerin bir uçtan diğer uca iletimi sırasında, üçüncü bir taraf ya da saldırgan tarafından değiştirilmediğinin garanti altına alınması büyük önem arz etmektedir. Bu durum yalnızca veri iletimiyle sınırlı kalmamakta, aynı zamanda verilerin depolanması ve daha sonraki zamanlarda bu verilere müdahale edilip edilmediğinin tespiti açısından da geçerliliğini korumaktadır. Söz konusu güvenlik gereksinimlerine yönelik olarak kriptografi bilimi çeşitli çözümler üretmiştir. Bu çözümler arasında Mesaj Doğrulama Kodları (Message Authentication Codes- MAC) ve Özetleme Fonksiyonları (Hash Functions) öne çıkmaktadır.

Bu çalışma kapsamında, blok zinciri teknolojisinde doğrudan kullanılmayan Mesaj Doğrulama Kodları detaylı biçimde ele alınmamıştır. Bu konuya ilgi duyan araştırmacılar

için [20] [21] kaynakları önerilmektedir. Buna karşın, özetleme fonksiyonları, blok zinciri teknolojisinin sunduğu değiştirilemezlik (immutability) gibi birçok temel özelliğin gerçekleşmesini mümkün kılan en önemli kriptografik yapı taşlarından biridir.

Özetleme fonksiyonları, matematiksel olarak iki küme arasında tanımlanmış fonksiyonlardır. Tanım kümesi, dijital verilerin modern bilgisayarların anlayabileceği biçimde ifade edildiği bit dizilerinin oluşturduğu küme olarak tanımlanmaktadır. Başka bir deyişle, $\{0,1\}^*$ biçiminde gösterilen bu kümede, her uzunluktan bit dizileri (0 ve 1'lerden oluşan diziler) yer alabilir. Bu dizilerin uzunluğu üzerinde herhangi bir kısıtlama bulunmamaktadır. Değer kümesi ise, belirli bir uzunluğa sahip sabit bit dizilerinden oluşur. Yani, bir H özetleme fonksiyonu aşağıdaki biçimde ifade edilebilir:

$$H: \{0,1\}^* \rightarrow \{0,1\}^n$$

Bu gösterimde, $H(x) = y$ eşitliğinde y değeri, x mesajının özet değeri (hash değeri) olarak adlandırılmaktadır. Böylece, herhangi bir uzunluktaki bir mesaj, sabit uzunlukta bir özet değeri ile temsil edilebilmektedir. Ancak, tanımı gereği özetleme fonksiyonları bire bir fonksiyonlar olmadığından farklı mesajlar aynı özet değerine sahip olabilmektedirler. Bu durum, çakışma (collision) olarak adlandırılır. Dolayısıyla, bir özetleme fonksiyonunun kriptografik amaçlarla kullanılabilmesi için yalnızca yukarıdaki matematiksel tanımı sağlaması yeterli olmamaktadır. Aşağıda sıralanan ek özelliklere de sahip olması gerekmektedir:

- **Öngörüntü Direnci (Pre-image Resistance):** $H(x) = y$ şeklindeki bir eşitlikte verilen y değeri için x değerinin pratik bir şekilde belirlenememesidir. Yani verilen bir özet değerinin hangi mesaja ait olduğunu belirlemenin yeterince zor olması durumudur. Bu özellik özetleme fonksiyonlarının tek-yönlü fonksiyonlar olarak adlandırılmasını da beraberinde getirmektedir.
- **İkincil Öngörüntü Direnci (Second Pre-image Resistance):** $H(x) = y$ şeklinde bir eşitlikte verilen y değeri için $H(x') = y$ olan bir başka x' değerinin pratik bir

şekilde belirlenememesidir. Yani özet değeri bilinen bir mesaj için aynı özet değerine sahip bir başka mesajın belirlenmesinin zor olması durumudur.

- **Çakışma Dayanıklılığı (Collision Resistance):** $H(x) = H(x')$ eşitliğini sağlayacak iki farklı x ve x' değerinin pratikte bir şekilde bulunamamasıdır. Yani aynı özet değerine sahip iki farklı mesajın belirlenmesinin zor olması durumudur.

Yukarıda sıralanan üç özelliği sağlayan özetleme fonksiyonları, kriptografik açıdan güvenli olarak kabul edilmektedir. Ek olarak, bir özetleme fonksiyonunun kullanışlı olabilmesi için verimli çalışması ve çıktılarının girişten bağımsız, sözde rastgele (pseudo-random) davranışlar sergilemesi de beklenmektedir. Bu bağlamda daha ayrıntılı bilgi için [21] kaynağı incelenebilir.

Özetleme fonksiyonları, birçok farklı alanda da yaygın olarak kullanılmaktadır. Bu alanlardan bazıları aşağıda sıralanmıştır:

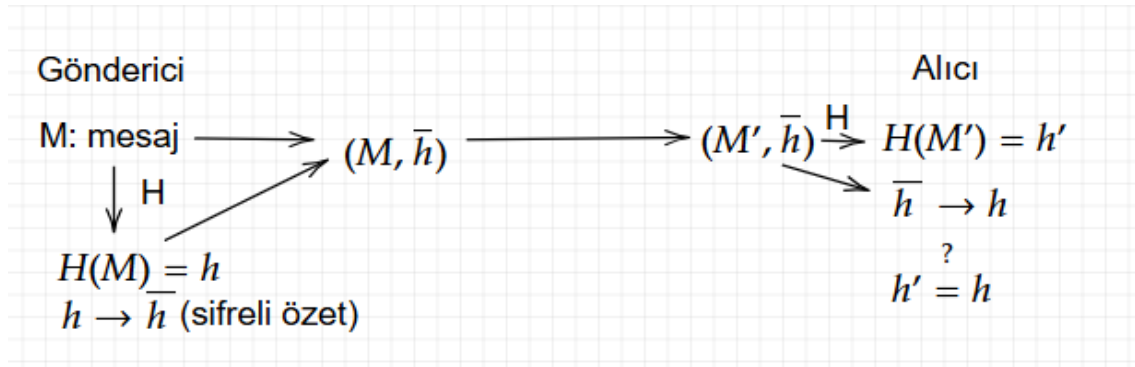
- **Parola Depolama:** Kullanıcı şifrelerinin doğrudan değil, özet değerleri ile veri tabanlarında saklanması yoluyla güvenlik sağlanmaktadır.
- **Rastgele Sayı Üretimi:** Çıktılarının sözde rastgele olması sayesinde, özetleme fonksiyonları bu alanda da işlevsel olarak kullanılmaktadır.
- **Dijital İmzalar:** İlerleyen bölümlerde ayrıntılı biçimde ele alınacağı üzere, dijital imza algoritmalarında büyük boyutlu mesajların özet değerlerinin imzalanması, işlem verimliliği açısından önem taşımaktadır.
- **Veri Bütünlüğü:** Blok zinciri teknolojilerinde özetleme fonksiyonlarının en yaygın kullanım alanlarından biri, veri bütünlüğünün sağlanmasıdır. Bu konu, bir sonraki alt bölümde daha ayrıntılı biçimde ele alınacaktır.

3.1.1. Özetleme fonksiyonlarının veri bütünlüğü sağlama rolü

Özetleme fonksiyonları, bir dijital verinin dijital parmak izi olarak da değerlendirilebilir. Bu bağlamda, özetleme fonksiyonlarının bir mesajın bütünlüğünü doğrulamak amacıyla kullanılması, bu bakış açısının en doğal ve işlevsel sonuçlarından biri olmaktadır.

Bu durumu daha iyi anlamak için, herhangi bir gizlilik beklentisinin olmadığı bir senaryo ele alınabilir. Örneğin, bir kullanıcı başka bir kullanıcıya bir mesaj göndermek istemektedir. Ancak bu iletişim, güvenli olmayan bir kanal (örneğin açık bir ağ) üzerinden gerçekleşmektedir. Bu durumda, mesajın üçüncü kişiler tarafından okunması veya değiştirilmesi mümkün olmaktadır. Senaryo gereği, mesajın okunabilir olması bir risk oluşturmazsa da, mesajın değiştirilip değiştirilmediğinin tespiti önem taşımaktadır. Alıcının, mesajın orijinal hâliyle ulaşmış ulaşmadığını anlaması, ek bir doğrulama mekanizması olmaksızın mümkün olmayacaktır. Bu soruna çözüm olarak, kriptografik açıdan güvenli bir özetleme fonksiyonu (örneğin H) kullanılabilir. Gönderici, iletmek istediği M mesajı için özetleme fonksiyonu $H(M) = h$ hesaplamasını gerçekleştirir. Daha sonra, gönderici M mesajını ve bir şekilde şifrelenmiş özet değeri h 'yi alıcıya iletacaktır. Alıcı ise kendisine ulaşan mesaj M' için özet değeri $H(M') = h'$ hesaplamalıdır. Göndericinin iletmediği şifreyi çözerek elde ettiği h değeri ile kendisinin elde ettiği h' değerinin eşit olup olmadığını kontrol etmelidir. Eğer $h = h'$ ise, mesajın gönderici tarafından gönderilen orijinal mesajla çok yüksek bir olasılıkla aynı olduğu sonucuna varılır. Ancak iki özet değeri farklı ise, mesajın iletişim sırasında bir şekilde değiştirildiği, yani veri bütünlüğünün bozulduğu anlaşılmaktadır. Aşağıda bu süreci görselleştiren bir şema sunulmaktadır:

Şekil 4: Özetleme fonksiyonları ile mesajın değişmediğinin kontrolü

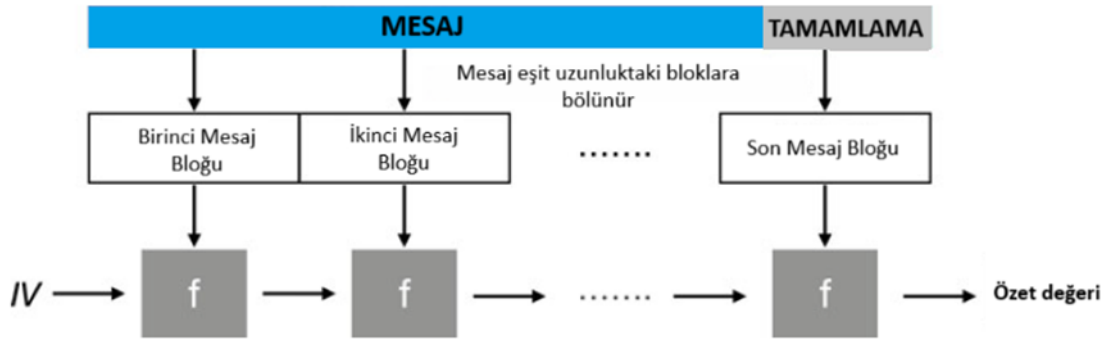


Gizliliğin de ön planda olduğu durumlarda ilgili kriptografik mekanizmalar yardımıyla bu şemanın başka varyasyonları da senaryo gereksinimlerine göre tasarlanabilmektedir. Detaylar için [22] kaynağı incelenebilir.

3.1.2. SHA-256 ve bazı diğer özetleme fonksiyonları

Özetleme fonksiyonları tanımlanırken yaygın olarak kullanılan yapılardan biri de Merkle-Damgard yapısıdır [23]. Bu yapının formu aşağıdaki diyagramda verilmektedir.

Şekil 5: Merkle-Damgard yapısı



Literatürde çok sayıda özetleme fonksiyonu algoritması tanımlanmıştır. Günümüzde en yaygın biçimde kullanılan özetleme fonksiyonu ailesi ise SHA (Secure Hash Algorithm)-2 ailesidir [24]. Bu ailenin tasarımında, önceki nesil özetleme fonksiyonları olan MD5 ve SHA-1 algoritmaları referans alınmıştır [21]. Bahsi geçen tüm bu algoritmalar, Merkle-Damgard yapısını kullanmaktadır.

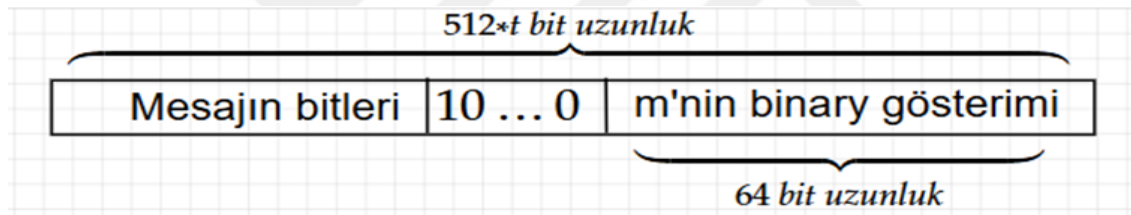
SHA-2 ailesi, 2001 yılında Amerika Birleşik Devletleri Ulusal Güvenlik Ajansı (NSA) tarafından standartlaştırılmıştır. Aile, altı farklı özetleme fonksiyonu içermektedir ve günümüzde pek çok alanda güvenlik amacıyla yaygın biçimde kullanılmaktadır. Bu altı algoritma arasından özellikle SHA-256 ve SHA-512, sırasıyla 256 bitlik ve 512 bitlik sabit uzunlukta çıktı üreten iki özetleme fonksiyonu olarak öne çıkmaktadır. Bu çalışmada, Bitcoin protokolünün temel bileşenlerinden biri olması nedeniyle, SHA-256 algoritması detaylı biçimde ele alınacaktır.

SHA-256, 256 bit uzunluğunda sabit bir özet değeri üreten bir özetleme fonksiyonu algoritmasıdır. Daha önce de belirtildiği gibi, bu algoritma değişken uzunlukta mesajlar alabilir. Ancak algoritmanın yapısı gereği, işlenebilecek maksimum mesaj uzunluğu 2^{64} bit ile sınırlandırılmıştır. Detaylı teknik açıklamalar için [25] standardına başvurulabilir. SHA-256 algoritması da diğer SHA-2 algoritmaları gibi Merkle-Damgard yapısı kullanılarak

tasarlanmıştır. Bu yapı kapsamında, orijinal mesaj belirli boyutlardaki bloklara ayrılır. SHA-256 algoritmasında her blok, 512 bit uzunluğundadır. Ancak, mesajların uzunluğu her zaman tam olarak bu blok boyutuna uymayabileceğinden, son bloğun oluşturulabilmesi için padding (bit tamamlama) işlemi uygulanması gerekmektedir.

Bu ön işleme adımının amacı, mesajın uygun biçimde parçalara ayrılabilmesini ve her parçanın algoritma içinde işlenebilir hâle gelmesini sağlamaktır. Tamamlama işleminin ardından, mesajın tüm blokları sırasıyla işlenerek nihai 256 bitlik özet değeri elde edilmektedir.

Tamamlama: $m < 2^{64}$ bit uzunluğunda bir M mesajın bit uzunluğu 512'ye bölünebilmesi için aşağıdaki şekilde bir tamamlama yapılmaktadır.



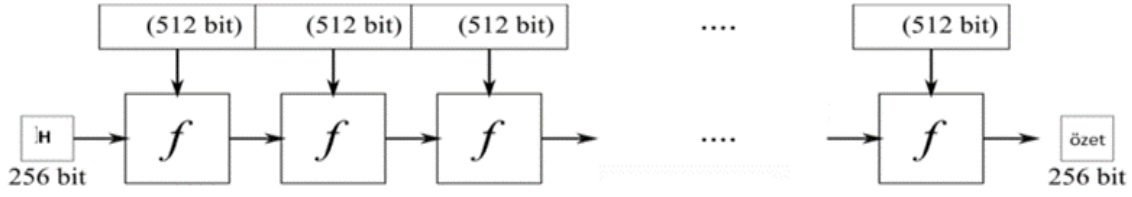
Sonuç olarak mesaj artık t tane 512 bit uzunluğunda bloğa bölünebilir hale gelmiş olacaktır.

Mesajın bloklarını $M_1, M_2, \dots, M_t$ şeklinde gösterelim. Merkle-Damgard yapısı gereği, algoritmanın başlatılabilmesi için 256 bit uzunluğunda bir başlangıç vektörüne ihtiyaç duyulmaktadır. Bu başlangıç vektörü, 32 bitlik sekiz adet kelimedenden oluşmaktadır. Her bir kelimenin onaltılık (hexadecimal) gösterimi aşağıda sırasıyla verilmiştir:

$$H_0 := 6A09E667, \quad H_1 := BB67AE85, \quad H_2 := 3C6EF372, \quad H_3 := A54FF53A$$

$$H_4 := 510E527F, \quad H_5 := 9B05688C, \quad H_6 := 1F83D9AB, \quad H_7 := 5BE0CD19$$

Temel olarak yapı aşağıdaki diagram ile gösterilebilir:



Şekilde görüleceği üzere her biri aynı şekilde tanımlanmış f fonksiyonu SHA-256 algoritmasının temelini oluşturmaktadır. Bu fonksiyon 8 tane 32 bitlik kelimeden oluşan (256 bit) bir girdi ile, 512 bitlik ilgili adımdaki mesaj bloğunu girdi olarak alarak 8 tane 32 bitlik kelimeden oluşan (256 bit) çıktı vermektedir. Son mesaj bloğu ile elde edilen çıktı ise mesajın özet değeri olarak elde edilir.

Şimdi bu f fonksiyonu ayrıntılı bir şekilde incelenecektir. Bu fonksiyon aynı işlevi kendi içerisinde 64 kere yapan tekrarlı bir yapıya sahiptir. Bu tekrarlara raunt adı verilir. Her bir raunt ilgili mesaj bloğunu bir şekilde kullanması gerekeceğinden ilk olarak ilgili mesaj bloğunu 64 rauntta kullanılacak şekilde düzenlemek gerekmektedir. Öncelikle 512 bitlik blok 32 bitlik 16 parçaya bölünmektedir. Bu parçalar yardımıyla 64 adet 32 bitlik ilgili raunda kullanılacak kelimeler elde edilmektedir. Bu işlem aşağıda tanımlanan şekilde yapılmaktadır:

W_0	...	W_{15}	W_{16}	...	W_i	...	W_{63}
-------	-----	----------	----------	-----	-------	-----	----------

Burada $W_0, \dots, W_{15}$ mesajın sırasıyla ilk 16 tane 32 bitlik kelimeleri olmak üzere, $i > 15$ için $\boxplus$ işlemi mod 2^{32} ye göre toplama işlemi olmak üzere,

$$W_i = W_{i-16} \boxplus \sigma_0(i) \boxplus W_{i-7} \boxplus \sigma_1(i)$$

şeklinde tanımlanmaktadır.

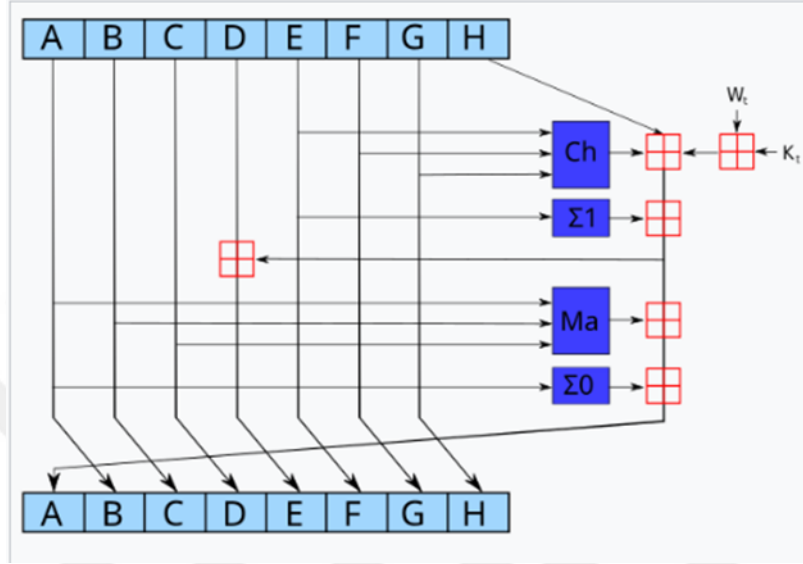
Burada $\oplus$, bit bit mod 2 ye göre toplama (XOR) işlemi, $R_j(\cdot)$ aldığı bit dizisini j birim sağa kaydırıp son j biti ilk j bitin yerine yazan operatör ve $S_j(\cdot)$ aldığı bit dizisini j bit sağa kaydırıp ilk j bitini 0 bitleri yapan operatör olmak üzere;

$$\sigma_0(i) = R_7(W_{i-15}) \oplus R_{18}(W_{i-15}) \oplus S_3(W_{i-15})$$

$$\sigma_1(i) = R_{17}(W_{i-2}) \oplus R_{19}(W_{i-2}) \oplus S_{10}(W_{i-2})$$

şeklinde tanımlanmış operatörlerdir. Böylelikle her bir 64 raunt için elimizde 64 adet 32 bitlik kelimelerden oluşan ve ilgili mesaj bloğundan elde edilmiş veriler oluşturulmuştur. Bir f fonksiyonunun bir rauntta yaptığı işin şemasını aşağıda gösterilmektedir.

Şekil 6 : Sha-256 Algoritmasında f fonksiyonunun bir raunt süreci



Kaynak: <https://en.wikipedia.org/wiki/SHA-2>

Burada A, B, C, D, E, F, G ve H değerleri 32 bitlik kelimeler raundun girdileri ve raunt sonunda ilgili raundun çıktılarıdır. K_t ise t . raundaki raunt sabitidir (32 bit). Bu bölüm sonunda bu sabitlerin değerleri verilecektir. Ayrıca, burada $\wedge$ bit bit “ve” operatörü, ve bir bit dizisi M için M' ise her bir bitin değerini alan operatörü göstermek üzere

$$Ch(E, F, G) = (E \wedge F) \oplus (E' \wedge G)$$

$$Ma(A, B, C) = (A \wedge B) \oplus (A \wedge C) \oplus (B \wedge C)$$

$$\Sigma_0(A) = R_2(A) \oplus R_{13}(A) \oplus R_{22}(A)$$

$$\Sigma_1(E) = R_6(E) \oplus R_{11}(E) \oplus R_{25}(E)$$

şeklinde tanımlanmış binary operatörlerdir. Tüm raundlar tamamlandıktan sonra ilgili f fonksiyonunun nihai çıktısı ise ilk raunda giren A, B, C, D, E, F, G ve H değerleri ile Son raundun çıktısı olan bu değerlerin mod 2^{32} ye göre toplanmasıyla elde edilen 32 bitlik değerler olacaktır. Mesajın bir bloğu için uygulanan bu işlemler Merke-Damgard yapısı gereği son mesaj bloğu tamamlana kadar uygulandıktan sonra en son adımdaki f

fonksiyonunun çıktısı olan 8 tane 32 bitlik kelime (256 bit dizisi) aynı sıralamayla yan yana yazılarak mesajın özet değeri hesaplanmış olacaktır. Aşağıda 64 raunt için kullanılan raunt sabitleri verilmiştir.

32 bitlik raunt sabitlerinin hexadecimal değerleri sırasıyla aşağıdaki gibidir:

428A2F98, 71374491, B5C0FBCF, E9B5DBA5, 3956C25B, 59F111F1, 923F82A4, AB1C5ED5, D807AA98, 12835B01, 243185BE, 550C7DC3, 72BE5D74, 80DEB1FE, 9BDC06A7, C19BF174, E49B69C1, EFBE4786, 0FC19DC6, 240CA1CC, 2DE92C6F, 4A7484AA, 5CB0A9DC, 76F988DA, 983E5152, A831C66D, B00327C8, BF597FC7, C6E00BF3, D5A79147, 06CA6351, 14292967, 27B70A85, 2E1B2138, 4D2C6DFC, 53380D13, 650A7354, 766A0ABB, 81C2C92E, 92722C85, A2BFE8A1, A81A664B, C24B8B70, C76C51A3, D192E819, D6990624, F40E3585, 106AA070, 19A4C116, 1E376C08, 2748774C, 34B0BCB5, 391C0CB3, 4ED8AA4A, 5B9CCA4F, 682E6FF3, 748F82EE, 78A5636F, 84C87814, 8CC70208, 90BEFFFA, A4506CEB, BEF9A3F7, C67178F2

Bu raunt sabitleri mantıksal olarak ilk 64 asal sayının küp köklerinin virgülden sonraki basamaklarının binary temsillerinin ilk 32 biti olarak türetilmişlerdir. SHA-2 ailesinin diğer algoritmaları ve detaylar için [25] incelenebilir.

Günümüzde bir başka standart algoritmalar ailesi olan SHA-3 algoritması da sıklıkla kullanılmaktadır [26]. Bu algoritma önceki standartların aksine Merkle-Damgard yapısı yerine sünger yapısını kullanmaktadır. Ayrıca bilinen bir simetrik şifreleme algoritması yardımı ile inşa edilebilen Davis-Meyer özetleme fonksiyonu da sıklıkla kullanılmaktadır [27].

Kriptografinin temellerinden olan özetleme fonksiyonları blok zinciri teknolojilerinde sıklıkla kullanılmaktadır. Hangi özetleme fonksiyonunun kullanıldığı, ilgili blok zinciri teknolojinin tasarımına göre değişiklik göstermektedir. Örneğin Bitcoin SHA-256 özetleme fonksiyonunu kullanırken, Ethereum Keccak-256 özetleme fonksiyonunu kullanmaktadır.

3.1.3. Özetleme fonksiyonlarının güvenliği

Güvenlik amacıyla tasarlanan her bir kriptografik yapının, güvenlik analizi ve güvenlik seviyeleri, sistemin güvenilirliği açısından büyük önem taşımaktadır. Bu bağlamda, kriptolojinin bir alt disiplini olan kriptoteorinin, söz konusu analizleri gerçekleştiren ve olası zayıflıkları ortaya koyan bilim dalı olarak tanımlanabilir.

Bu doğrultuda, özetleme fonksiyonlarının güvenliği de üzerinde dikkatle durulması gereken bir konudur. Kriptografik anlamda güvenli olarak nitelendirilen bir özetleme fonksiyonunun güvenliği, genellikle çakışma dayanıklılığı (collision resistance) seviyesiyle ölçülmektedir. Bir özetleme fonksiyonunun çakışma dayanıklılığına ilişkin en temel değerlendirme ise literatürde doğum günü paradoksu [28] olarak bilinen olasılık teorisi temelli bir problemin genellemesi ile yapılmaktadır.

Problem: Bir topluluktan en az kaç kişi seçilmelidir ki bu seçilen kişilerden ikisinin doğum gününün aynı olma olasılığı $\frac{1}{2}$ den fazladır?

Bu sorunun cevabı şaşırtıcı bir şekilde 23 olarak bulunur hatta bu olasılık 77 kişilik grupta %99 ların üzerine çıkmaktadır. Şimdi bu problemi şu şekilde ifade edelim:

Problem: A , eleman sayısı yeterince büyük bir küme ve B ise T elemanlı bir küme olsun.

$$f: A \rightarrow B$$

şeklinde tanımlanan bir f fonksiyonu için kaç farklı girdi denemeliyiz ki bunlardan en az ikisinin çıktı değeri %50 den yüksek olasılıkla aynı olsun?

Aslında bu problem “bir özetleme fonksiyonu için kaç aramada %50 ihtimalle çakışma bulabiliriz?” ifadesine denk olmaktadır. Şimdi bunun için bir çözüm sunalım. Kabul edelim ki bu sayı n olsun o halde doğrudan olasılık hesaplaması ile bu n denemenin her birinin farklı

değere sahip olma olasılığı hesaplanarak (problem gereği bu olasılıkta yaklaşık %50 olmalı) aşağıdaki gibi formülize edilebilir.

$$1 \left(\frac{T-1}{T} \right) \left(\frac{T-2}{T} \right) \dots \left(\frac{T-n+1}{T} \right) \approx \frac{1}{2}$$

$$\left(1 - \frac{1}{T} \right) \left(1 - \frac{2}{T} \right) \dots \left(1 - \frac{n-1}{T} \right) \approx \frac{1}{2}$$

Analiz teorisindeki e^x fonksiyonun Maclaurin seri açılımı [29] düşünüldüğünde yukarıdaki ifadenin sol tarafı yaklaşık olarak (yeterine büyük T değerleri için) aşağıdaki gibi yazılabilir:

$$e^{-1/T} e^{-2/T} \dots e^{-(n-1)/T} = e^{-\frac{1}{2T}(n^2-n)} \approx e^{-\frac{n^2}{2T}}.$$

Bu değer yaklaşık $\frac{1}{2}$ olabilmesi için

$$n^2 \approx 2T \ln 2 \Rightarrow n \approx 1,177\sqrt{T}$$

olarak hesaplanmaktadır. Büyük değerler için 1,177 sayısı ihmal edilebileceğinden seçimin yaklaşık $\sqrt{T}$ kadar deneme olması durumunda iki çıktının değerinin aynı olma olasılığı %50 olacaktır. Özetleme fonksiyonları gereği düşünersek sabit çıktı bit uzunluğu N olan bir özetleme fonksiyonu için (2^N tane farklı çıktı) çakışma bulma olasılığı $2^{N/2}$ denemede %50 olacaktır ve buda yeterince yüksek bir orandır. SHA-256 özelinde düşünersek %50 ihtimalle çakışma bulmak için 2^{128} farklı mesaj için özet değeri hesaplanmalıdır ki bu durumda günümüz bilgisayarlarıyla anlamlı süreler içerisinde olası değildir.

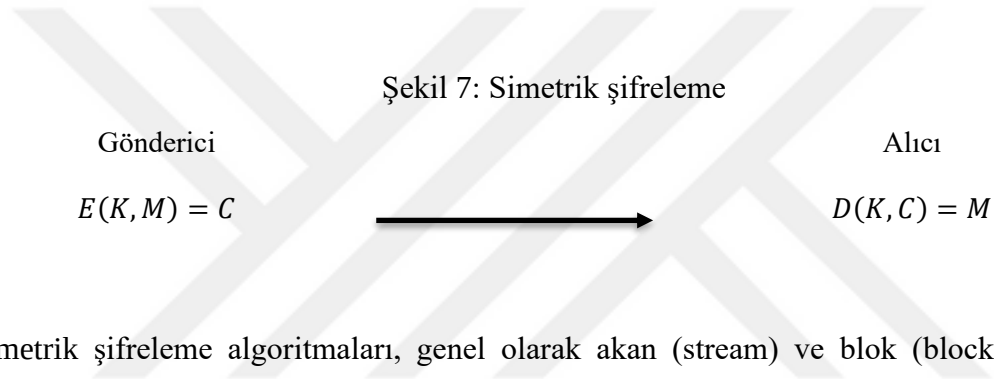
3.2. Simetrik Kriptografi

Kriptografi biliminin temel amaçlarından bir diğeri de gizliliği sağlamaktır. Kriptografik sistemler, iletişimin gerçekleştiği kanalın her zaman güvensiz olduğu varsayımı üzerine inşa edilmiştir. Bu varsayım, üçüncü şahısların iletişimi izleyebileceği veya mesajlara müdahalede bulunabileceği anlamına gelir. Böyle bir ortamda, göndericinin mesajı doğrudan alıcıya iletmesi, mesaj içeriğinin kolayca açığa çıkmasına neden olabilir.

Kriptografide gizlilik, şifreleme (encryption) ve deşifreleme (decryption) mekanizmaları aracılığıyla sağlanır. Gönderici, iletmek istediği mesajı bir şifreleme fonksiyonu aracılığıyla

şifreler ve bu şifreli veriyi alıcıya iletir. Alıcı ise, deşifreleme fonksiyonu yardımıyla gelen veriyi çözümler ve orijinal mesaja ulaşır.

Burada dikkat edilmesi gereken önemli bir nokta, şifreleme işleminin arkasındaki fonksiyonun kendisinin gizli olmamasıdır. Gizliliği sağlayan temel unsur, yalnızca gizli bilgi olarak tanımlanan ve anahtar (key) olarak adlandırılan veridir. Şayet hem gönderici hem de alıcı, aynı gizli anahtarı kullanarak mesajı şifreliyor ve çözümlüyorsa, bu tür sistemlere simetrik şifreleme mekanizmaları adı verilir. M orijinal mesajı, E şifreleme fonksiyonunu, D deşifreleme fonksiyonunu, K gizli anahtar ve C şifreli mesajı göstermek üzere bir simetrik şifreleme yapısı aşağıdaki gibi özetlenebilir.



Simetrik şifreleme algoritmaları, genel olarak akan (stream) ve blok (block) şifreleme algoritmaları olmak üzere ikiye ayrılır. Günümüzde özellikle blok şifreleme yapıları (AES gibi) daha yaygın olarak kullanılmaktadır.

Bununla birlikte, blok zinciri teknolojileri, doğaları gereği şeffaflık ilkesine dayandıkları için gizlilik sağlamak gibi bir zorunluluk taşımazlar. Bu nedenle, blok zinciri uygulamalarında şifreleme mekanizmaları yaygın olarak kullanılmaz. Bu durum, kriptografi ile blok zinciri teknolojileri arasında sıkça karıştırılan yaygın bir yanlış anlamının da temelini oluşturmaktadır. Örneğin, Bitcoin teknolojisi, isim olarak “kriptografi” ile ilişkilendirilmiş olsa da, veri şifreleme mekanizması içermez.

Bu nedenle, bu tez kapsamında simetrik şifreleme yapıları ayrıntılı olarak ele alınmayacaktır. Konuya ilgi duyan araştırmacılar için bu kaynakların [21] [30] incelenmesi önerilmektedir.

3.3. Asimetrik Kriptografi

Simetrik şifreleme mekanizmaları, temel gizlilik ihtiyaçlarını karşılamakla birlikte, bazı yapısal sorunlar barındırmaktadır. Bu sorunların başında iki temel problem öne çıkmaktadır. Birincisi, bir kullanıcının iletişim kurmak istediği kişi sayısı arttıkça, sahip olması gereken gizli anahtar sayısı da aynı oranda artmaktadır. Başka bir ifadeyle, her iki kullanıcı çifti arasında farklı bir gizli anahtar kullanılmalıdır. Bu durum, özellikle büyük ölçekli sistemlerde aşırı sayıda anahtarın üretilmesini ve güvenli şekilde saklanmasını gerektirir. Anahtarların yönetimi, saklanması ve güncellenmesi ciddi bir yük oluşturur. İkinci problem ise, taraflar arasında paylaşılan gizli anahtarın üçüncü kişiler tarafından ele geçirilemeyecek biçimde iletilmesidir. Ancak, iletişim kanalının güvensiz olduğu varsayımı altında bu anahtarın paylaşımı başlı başına bir güvenlik riski oluşturur. Güvenli anahtar iletimi sağlanmadan, tüm sistemin güvenliği tehlikeye girebilir.

Ek olarak, simetrik şifreleme mekanizmaları, gizlilik ve kısmen de veri bütünlüğü gibi güvenlik amaçlarına hizmet ederken, kimlik doğrulama (authentication) konusunda yeterli çözüm sunamamaktadır.

Bu sorunlara çözüm üretmek ve simetrik şifrelemenin eksik yönlerini tamamlamak amacıyla asimetrik kriptografi geliştirilmiştir. Asimetrik şifreleme yapıları ilk kez Whitfield Diffie ve Martin Hellman tarafından 1976 yılında yayımlanan öncü çalışmayla kriptografi literatürüne kazandırılmıştır [2]. Bu çalışmada, bir simetrik şifrelemede kullanılan anahtarın her iki taraf içinde ortak bir şekilde hesaplamasına yönelik bir mekanizma sunulmuştur. Blok zinciri teknolojisinde özellikle verilerin transferinde dijital imzalama ve kullanıcı kimliğini temsil eden açık adreslerin oluşturulması asimetrik kriptografi araçları kullanılarak gerçekleştirilmektedir. Dolayısıyla özetleme fonksiyonlarının ardından, blok zinciri teknolojisinin en önemli kriptografik bileşenlerinden birinin asimetrik kriptografi olduğu söylenebilir.

3.3.1. Asimetrik Kriptografinin Tanımı

Asimetrik kriptografi (ya da diğer adıyla açık anahtarlı kriptografi), simetrik şifrelemeden farklı olarak iletişimi tek bir anahtar üzerinden gerçekleştirmez. Bu sistemlerde, her bireyin,

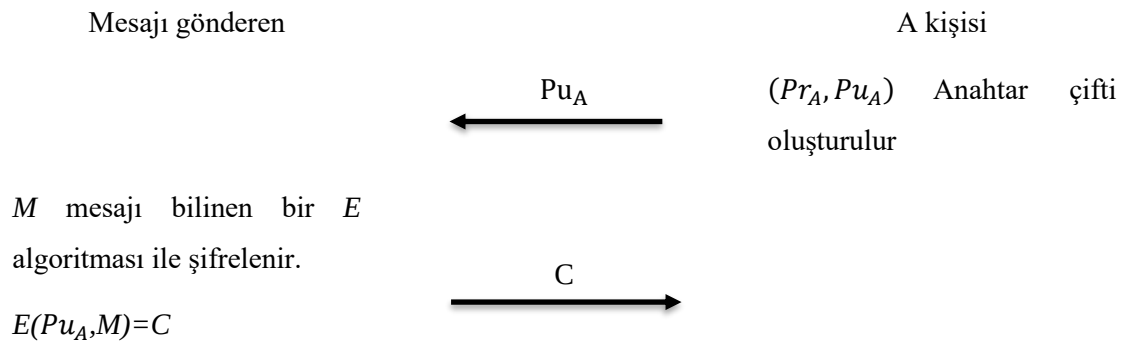
yalnızca kendisine ait ve gizli tutması gereken bir özel (gizli) anahtarı ve herkesin erişebileceği bir açık anahtarı bulunmaktadır. Bu yapı sayesinde, her birey yalnızca tek bir gizli anahtar saklamak zorundadır. Bu durum, simetrik şifrelemede karşılaşılan anahtar sayısı ve bunların depolama ihtiyacının artması problemine etkili bir çözüm sunmaktadır. Asimetrik şifrelemelerde ihtiyacımız olan bazı kavramlar aşağıda yer almaktadır.

M : Mesaj, C : Şifreli Mesaj, E : şifreleme fonksiyonu, D : deşifreleme fonksiyonu, Pr_A : A kişinin gizli anahtarı, Pu_A : A kişinin açık anahtarı.

3.3.2. Şifreleme ve Şifre Çözme

İlk adımda, A kişisi kendisine ait açık ve gizli anahtar çiftini üretmektedir. Bu işlem, kullanılan şifreleme algoritmasına bağlı olarak çeşitli matematiksel yapılar temelinde gerçekleştirilir. Anahtar üretimi sürecinin detayları ilerleyen alt bölümlerde ele alınacaktır. A kişinin oluşturduğu açık anahtar, sistemdeki diğer bireylerin erişimine sunulur (örneğin bir açık anahtar dizini aracılığıyla). Gizli anahtar ise yalnızca A kişinin erişiminde kalmalıdır. A kişisine gizli bir şekilde mesaj göndermek isteyen bir kişi mesajı A kişinin açık anahtarı ile şifreleyerek iletmektedir. Bu şifreli mesajı alan A kişisi de kendi gizli anahtarını kullanarak deşifreleme işini yaparak orijinal mesajı elde etmektedir. Görüleceği üzere şifreleme ve deşifreleme fonksiyonlarının anahtar girdileri, simetrik şifrelemenin aksine aynı değildir. “Asimetrik” kavramının kullanılması bu sebeptendir. Aşağıda ki şema ile bu anlatılanlar özetlenmektedir.

Şekil 8: Asimetrik şifreleme



C şifreli mesajı bilinen D algoritması ile çözülür.

$$D(Pr_A, C) = M$$

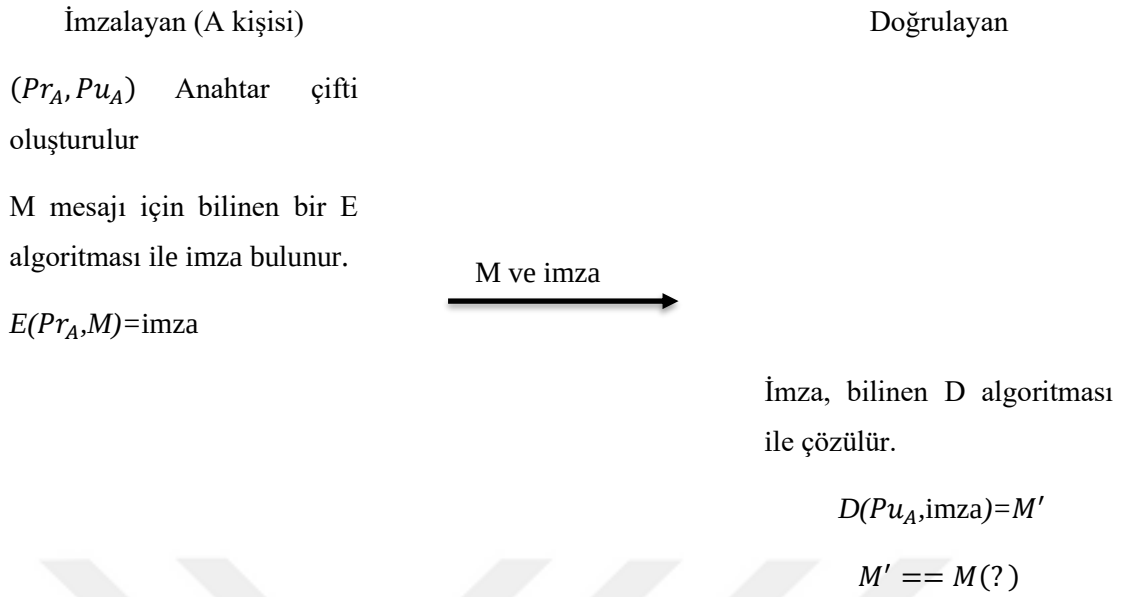
Yukarıdan da görüleceği üzere E ve D fonksiyonlarına birbirlerinin doğrudan tersleri değildirler. İlerleyen alt bölümlerde bilinen bazı önemli asimetrik şifreleme mekanizmalarından detaylıca bahsedilecektir.

3.3.3. Dijital İmzalar

Asimetrik kriptografi yukarıda verdiğimiz gizlilik amacına hizmet etmesinin yanında kimlik doğrulama amacına da hizmet etmektedir. Bu bölümde öncelikle dijital imzanın ne demek olduğundan bahsetmek doğru bir yaklaşım olacaktır. Gerçek hayatta insanlar kimliklerini doğrulamak için yaptıkları işlemlerde fiziksel imzalarını kullanmakta ve karşı tarafta kişilerin attıkları bu imzaya bakarak onların doğru kişi olup olmadığını doğrulayabilmektedir. Fakat dijitalleşen dünyada fiziksel imzanın karşılığı olamayacağı için dijital imza, bir kişinin yaptığı bir iş için (gönderdiği mesaj vs.) alıcıya sunduğu bir dijital kimlik ispatı olmaktadır. Alıcı, bu ispatın doğruluğunu belirli kriptografik algoritmalar yardımıyla kontrol edebilmektedir.

Bir asimetrik kriptografi mekanizması var ise (şifreleme ve deşifreleme algoritmaları) bu mekanizma bir dijital imza metodu olarak da kullanılabilir. Bir A kişisi göndereceği bir M mesajı için kendi gizli anahtarı ile şifreleme yaparak (A kişinin gizli bilgisi eklenmiş oldu), karşı tarafa mesajı ve bu şifreyi göndermektedir. Alıcı ise A kişinin açık anahtarı ile imzayı deşifre ederek ve kendine gelen mesaj ile karşılaştırarak sonuca göre imzanın doğruluğuna ya da geçersizliğine karar verebilmektedir. A 'nın imzayı oluşturduğu aşamaya imzalama algoritması, alıcının imzanın geçerliliğine karar verdiği aşamaya ise doğrulama algoritması adı verilir. Aşağıdaki şemada bu senaryo özetlenmektedir.

Şekil 9: Dijital imza çalışma mekanizması



Yukarıda tasvir edilen mekanizmanın çeşitli dezavantajları bulunmaktadır. Örneğin çok uzun mesajlar için gönderilecek imza çok büyük olmaktadır. Bu problemin çözümü için imza mesaja değil mesajın özet değerine atılmaktadır. Bir diğer dezavantaj ise asimetrik kriptografik mekanizmaların yavaş olmasıdır. Dolayısıyla bunları dijital imza mekanizması olarak kullanmak verimli değildir. Buna çözüm olarak ise sadece dijital imzaya yönelik mekanizmalar geliştirilmiştir. Bu tür yapılardan ilerleyen alt bölümlerde bahsedilecektir. Detaylar için bu kaynaklar [21] [31] incelenebilir.

3.3.4. Asimetrik Kriptografide Anahtarlar

Daha önce de belirttiğimiz gibi, bir asimetrik kriptografi sisteminde her kullanıcının birbirinden farklı iki anahtarı bulunur: biri açık anahtar, diğeri ise gizli anahtardır. Bu iki anahtar, sistemin kullanılacağı amaca göre (örneğin, veri şifreleme ya da dijital imza üretme) farklı biçimlerde kullanılmaktadır. Ancak bu işlemlerin doğru ve güvenli çalışabilmesi için, bu anahtar çiftleri arasında matematiksel olarak tutarlı bir ilişki olması gerekmektedir. En kritik güvenlik gereksinimlerinden biri, açık anahtar herkes tarafından bilinse bile, bu anahtardan gizli anahtarın elde edilememesidir. Bu noktada, bazı matematiksel olarak zor problemler devreye girer. Asimetrik kriptografi sistemlerinin güvenliği, bu tür problemlerin çözümünün pratikte mümkün olmamasına dayanmaktadır. Aşağıda bu problemler içinde en önemlilerden olan “Çarpanlara Ayırma” ve “Ayrık Logaritma” problemleri incelenmektedir.

Çarpanlara Ayırma Problemi: Aritmetiğin Temel Teoremi gereği herhangi bir pozitif tamsayı tek türlü asal çarpanlarına ayrılabilir, yani bir $n \in \mathbb{N}$ için

$$n = p_1^{e_1} p_2^{e_2} \cdots p_r^{e_r}$$

olacak şekilde p_i asal sayıları ve e_i pozitif tam sayıları vardır. Çarpanlara ayırma problemi ise verilen bir n doğal sayısı için p_i asal sayılarını ve e_i pozitif tam sayılarını belirlemektir. Küçük n sayıları için bu problem kolay gibi gözükse de n sayısı büyüdükçe problem oldukça zorlaşmaktadır. Kriptografide bu problemin daha özel bir hali kullanılmaktadır. Bu problem ise iki farklı asalin çarpımı olan bir n doğal sayısı için, ($n = pq$, p ve q asal), bu asal sayıların bulunmasıdır. Bu probleme baktığımızda, p ve q bilinirken n yi belirlemek kolay bir iş iken, n bilinirken p ve q 'yu belirlemek oldukça zor bir iş olmaktadır. Bu tür problemler tek yönlü problemler olarak adlandırılmaktadır. Buna bir benzer problem olarak özetleme fonksiyonlarında ön görüntü bulma problemi örnek gösterilebilir.

Ayrık Logaritma Problemi: Kriptografide en yaygın kullanılan zor problemlerden biridir. Verilen bir m tamsayısı ve $g < m$ değeri için:

$$g^x = y \pmod{m}$$

eşitliğinde eğer y değeri biliniyor ise x değerini belirlemek, ayrık logaritma problemi olarak tanımlanmaktadır. Yine görüleceği üzere x bilinirken, y değerini hesaplamak oldukça kolay olurken tersine y bilinirken x 'i belirlemek çok zor bir problem olmaktadır. Hatta bazı durumlarda, bu eşitliğin çözümünün olup olmadığı bile net biçimde belirlenememektedir. Yaygın kriptografi uygulamalarında, bu problemin m 'nin bir asal sayı ve g nin ise mod m ye göre ilkel eleman olması yani,

$$g^{m-1} = 1 \pmod{m} \text{ fakat her } 1 \leq i \leq m-2 \text{ için } g^i \neq 1 \pmod{m}$$

şartını sağlayan g değerleri, durumundaki versiyonu kullanılmaktadır.

3.3.5. Bazı Önemli Asimetrik Kriptografi Algoritmaları

Bu alt bölümde literatürde bilinen ve yaygın olarak kullanılan bazı önemli asimetrik kriptografi mekanizmalarından bahsedilecektir. Bu algoritmalar hakkında daha fazla detay ve diğer algoritmalar için [20] [31] kaynakları incelenebilir.

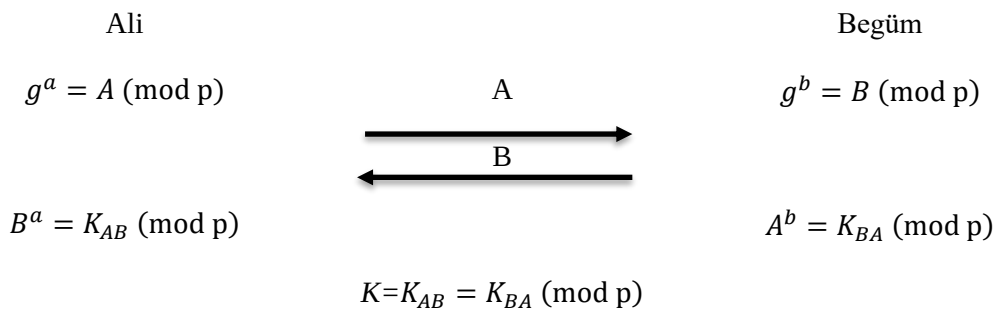
3.3.5.1. Diffie-Hellman Anahtar Anlaşma Protokolü

1976 yılında sunulmuş Diffie-Hellman anahtar anlaşma protokolü [32] asimetrik kriptografinin ortaya çıkmasını sağlamıştır. Bu protokol simetrik şifreleme mekanizmalarında problem olan ortak anahtar belirleme problemine güvenli bir çözüm sunmuştur. Protokolün güvenliği ayrık logaritma problemine dayanmaktadır. Protokolün çalışma sistematığı aşağıdaki şekildedir:

- Simetrik şifrelemede ortak anahtar belirlemeye çalışan Ali ve Begüm isimli iki kullanıcı olduğu varsayılmaktadır. Bu kullanıcılar öncelikle yeterince büyük bir p asal sayısı ve mod p ye göre bir ilkel eleman olan $g < p$ değerini belirleyeceklerdir.
- Daha sonra bu belirlenen parametreler yardımıyla, Ali ve Begüm sırayla 2 ve $p - 2$ değerleri arasında a ve b sayıları seçecekler Ali $g^a = A \pmod{p}$, Begüm ise $g^b = B \pmod{p}$ değerini hesaplayacaklardır. Burada, Ali'nin anahtar çifti $(Pr_A, Pu_A) = (a, A)$, Begümün anahtar çifti ise $(Pr_B, Pu_B) = (b, B)$ olarak elde edilecektir. Bu aşama anahtar üretme aşaması olarak adlandırılmaktadır.
- Ali Begüm'e A , değerini, Begüm ise Ali'ye B değerini gönderecektir.
- Ali $B^a = K_{AB} \pmod{p}$ değerini, Begüm ise $A^b = K_{BA} \pmod{p}$ değerini hesaplayacaktır.
- $K_{AB} = B^a = g^{ba} = g^{ab} = A^b = K_{BA} \pmod{p}$, olduğundan simetrik şifreleme için kullanılacak ortak anahtar $K = K_{AB} = K_{BA}$ belirlenmiş olmaktadır.

Aşağıdaki şemada yukarıda açıklanan protokol özetlenmiştir. Belirlenmiş yeterince büyük bir p asal sayısı ve mod p ye göre bir ilkel eleman olan $g < p$ değerleri için:

Şekil 10: Diffie-Hellman Anahtar Anlaşma Protokolü



Protokol, yalnızca ortak anahtarın belirlenmesi amacıyla kullanılmakta ve mesajların doğrudan şifrelenmesini sağlamamaktadır. Ancak bu yapı, daha sonra geliştirilen birçok asimetrik kriptografik mekanizmanın temelini oluşturmuştur.

Bu protokolde güvenlik, yalnızca A ve B değerlerine sahip olan bir saldırganın, a veya b gizli anahtarlarını hesaplayamaması durumuna dayanır. Bu işlem, pratikte mümkün olmayan ayrık logaritma problemini çözmeye karşılık gelir. Bu nedenle, yalnızca gizli anahtarlara sahip olan taraflar, ortak anahtarı hesaplayabilir.

3.3.5.2. RSA Şifreleme Mekanizması

Asimetrik kriptografide ilk şifreleme algoritması olan RSA, 1978 yılında Ron Rivest, Adi Shamir ve Leonard Adlema [3] n tarafından geliştirilmiştir. RSA algoritmasının güvenliği, yukarıda bahsettiğimiz, hesaplama açısından oldukça zor bir problem olan büyük sayıların çarpanlarına ayrılması problemine dayanmaktadır. Bu problem, özellikle çok büyük asal sayıların çarpımı olan bir sayının, asal çarpanlarına ayrılmasının pratikte mümkün olmaması üzerine kuruludur. Algoritmayı anlatmadan önce önemli ve ihtiyacımız olan bir tanıımı paylaşmak gerekli olacaktır.

Tanım 3.1 (Euler Totient fonksiyonu): $\phi: \mathbb{N} \rightarrow \mathbb{N}$, Euler Totient fonksiyonu bir $n \in \mathbb{N}$ girdisi için n 'den küçük n ile aralarında asal pozitif tam sayıların sayısını vermektedir, yani

$$\phi(n) := \{k \in \mathbb{N}: k < n \text{ ve } \text{ebob}(n, k) = 1\} \text{ kümesinin eleman sayısı}$$

olarak tanımlanmaktadır.

Bu fonksiyon hakkında daha detaylı ve teorik bilgiler için [33] kaynağının incelenmesi önerilmektedir.

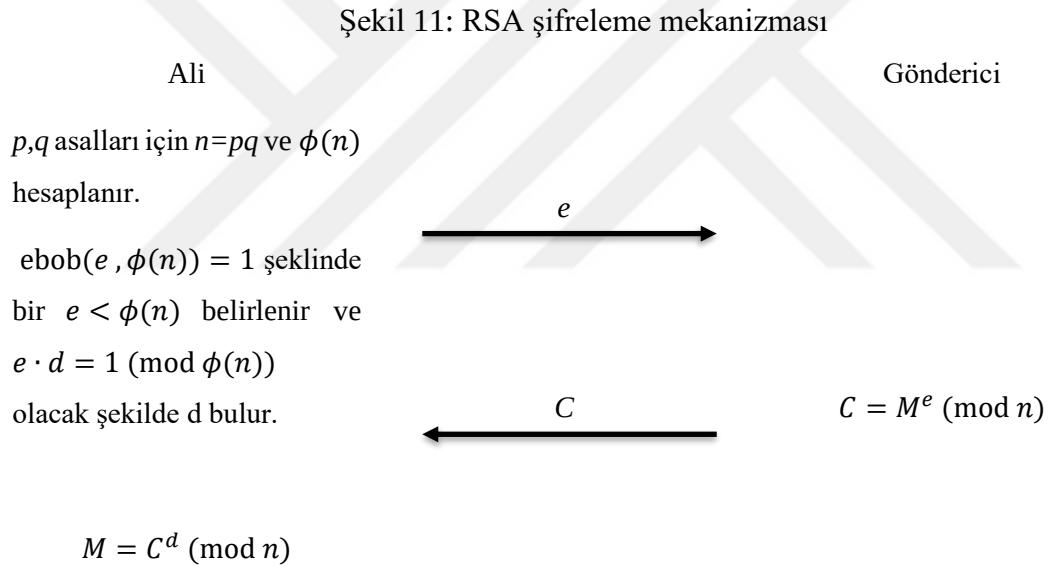
Artık RSA algoritmasını incelemeye başlayabiliriz.

- **Anahtar üretimi:** Ali, yeterince büyük iki farklı p ve q asal sayılarını belirleyerek ve $n = pq$ sayısını hesaplayacaktır. Daha sonra $\phi(n) = \phi(pq) = (p - 1)(q - 1)$ değerini hesaplayıp $\text{ebob}(e, \phi(n)) = 1$ olacak şekilde bir $e < \phi(n)$ sayısı

belirleyecektir. Bu bilgiler ve genişletilmiş Öklid algoritması yardımıyla [33] $e \cdot d = 1 \pmod{\phi(n)}$ olacak şekilde $d < \phi(n)$ sayısını bulunacaktır. Burada $p, q, \phi(n)$ ve d gizli anahtarlar, n ve e de açık anahtarlar olmaktadır.

- **Şifreleme:** Ali'ye M mesajını göndermek isteyen bir kişi $C = M^e \pmod{n}$ şifreli mesajını hesaplayarak bu değeri gönderecektir.
- **Deşifreleme:** Ali aldığı C şifreli mesajını gizli anahtarı d 'yi kullanarak $M = C^d \pmod{n}$ hesaplamasıyla orijinal mesaj haline getirecektir.

Burada deşifreleme fonksiyonunda orijinal M mesajının elde edilmesi Euler Teoremi'nin sonucudur [33]. RSA şifreleme mekanizması aşağıdaki şema ile özetlenmektedir.



Burada n yi bilen bir kişinin p ve q yu belirlemesi pratik değildir, dolayısıyla $\phi(n)$ de pratik bir şekilde belirlenememektedir. Yani e yi bilen bir kişi d 'yi pratik olarak hesaplayamaz. Gizli anahtarın güvenliği böylelikle çarpanlara ayırma problemine dayanmaktadır. Fakat çarpanlara ayırma problemini çözmeden eline bir şekilde $\phi(n)$ değerini geçirmiş kişi de p ve q değerlerini pratik olarak bulup sistemi kırabilmektedir. Dolayısıyla çarpanlara ayırma problemi ek bir bilgi ile pratik olarak çözülebilir hale gelmektedir. Bu tür zor problemlere arka kapılı tek yönlü problemler denilmektedir. Aşağıda $\phi(n)$ ve n bilinirken n 'nin p ve q asal çarpanlarının nasıl bulunacağı açıklanmaktadır.

$$\phi(n) = (p - 1)(q - 1) = pq - p - q + 1 = n - (p + q) + 1$$

olduğundan $(p+q)$ kolayca bulunabilecektir. İki pozitif sayının çarpımı ve toplamları biliniyorsa bu iki sayı kolaylıkla belirlenebilmektedir. Yani $\phi(n)$ bilindiğinde n yi çarpanlara ayırma problemi kolay bir probleme indirgenmiş olmaktadır.

Bölüm 3.3.3 te anlatıldığı üzere bir şifreleme algoritması olan RSA algoritması aynı zamanda bir dijital imzalama algoritması olarak ta kullanılabilir. Ayrıca şifrelemenin mesaj yerine ilgili simetrik şifrelemenin anahtarı için yapılması ile karşı tarafa iletilmesi yöntemiyle RSA algoritması bir anahtar değişimi algoritması olarak ta kullanılabilir.

RSA algoritmasının, doğal olarak bazı dezavantajları da mevcuttur. Bunlardan en önemlisi her bir kullanıcının kendi gizli ve açık anahtarlarını oluşturması gerekmektedir. Yani her biri iki farklı yeterince büyük asal sayı üretmelidir. Büyük bir sayının kesin olarak asal sayı olduğunu belirtmenin pratik bir yolu yoktur. Fakat geliştirilmiş bazı olasılıksal algoritmalarla yüksek olasılıkla asal sayı olup olmadığı kanısına varabiliriz. Bu algoritmalar yaygın olarak kullanılanı Miller-Rabin asallık testidir [34] [35]. Ayrıca yüksek güvenlik seviyesi için hem fazla işlem hem de yüksek kapasitede imza veya şifreli metin üretilecektir ve bu da düşük donanım veya hızın gerektiği ortamlarda kullanışlı olmadığını göstermektedir. Bu dezavantajın olmadığı yeni bir matematiksel model ile bir sonraki alt bölümde vereceğimiz mekanizma geliştirilmiştir.

3.3.5.3. ECC – Eliptik Eğri Şifreleme Mekanizması

RSA algoritmasının yüksek güvenlik seviyelerinde yavaşlık ve büyük boyutlu çıktı vermesi dezavantajını ortadan kaldırmak amaçlı yeni bir matematiksel aritmetik tanımlı küme yardımıyla bir kriptografik mekanizma geliştirilmiştir. Bu yapı geometrik objeler olan eliptik eğrilerin üzerindeki noktalar kümesi üzerinde tanımlanmış aritmetik ile inşa edilmiştir. Böyle bir sistem 1985 yılında birbirinden bağımsız olarak Neal Koblitz ve Victor Saul Miller adlarında iki matematikçi tarafından önerilmiş [36] [37] daha sonra geliştirilmiştir. Mekanizmanın detaylarını vermeden önce eliptik eğriler üzerine gerekli ön bilgileri verelim.

Öncelikler geometrik açıdan da önemli olan eliptik eğrinin tanımını verelim.

Tanım 3.2: K bir cisim olmak üzere, $a, b \in K$ için $y^2 = x^3 + ax + b$ denklemini sağlayan (x, y) ikililerinin $K \times K$ düzlemindeki kümesine K üzerinde bir eliptik eğri denir. Yani

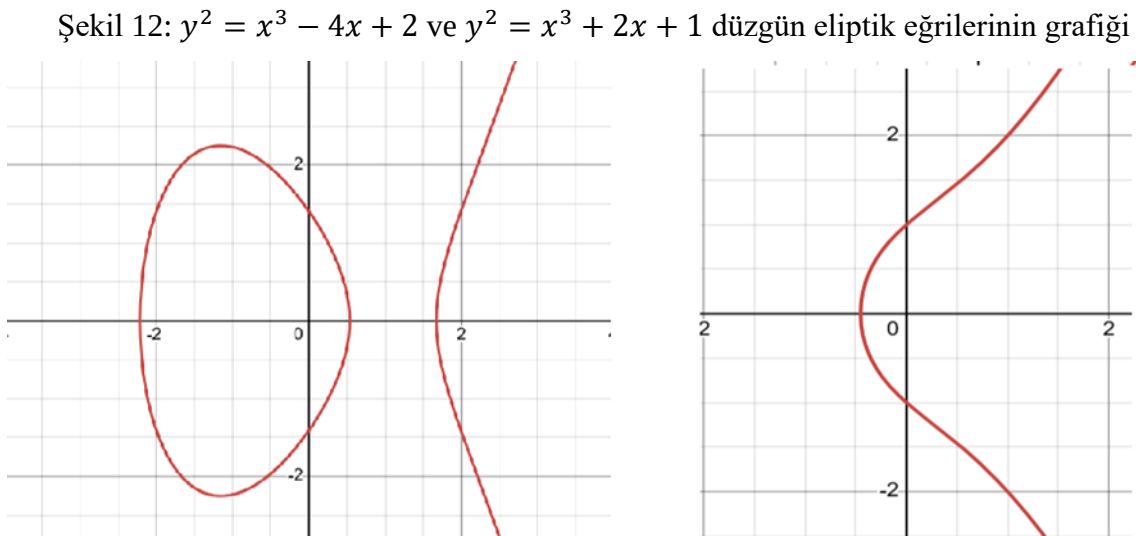
$$E = \{(x, y) \in K \times K : y^2 = x^3 + ax + b\}$$

kümesi eliptik eğri olarak adlandırılır.

Burada dikkat edilmelidir ki geometrik anlamda eliptik eğri daha genel formdadır fakat gerekli kaydırma ve döndürme işlemleri ile Tanım 3.2 de verilen form elde edilebilir bu forma özel olarak Weirstrass formu denir.

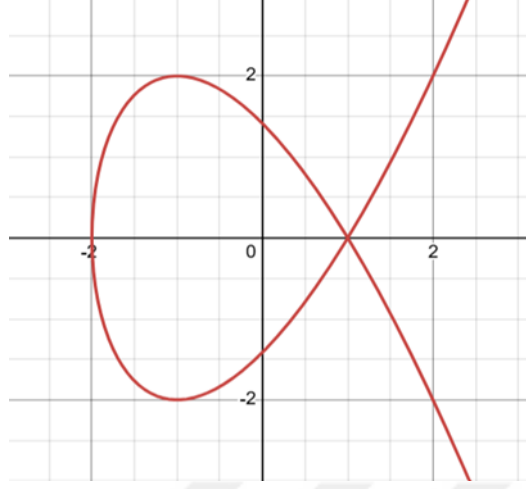
$y^2 = x^3 + ax + b$ formunda verilen bir eliptik eğri için eğer $\Delta = -16(4a^3 + 27b^2)$ diskriminant değeri sıfırdan farklıysa o eğriye düzgün eğri, sıfır ise tekil eğri adı verilir. Üzerinde tanımlayacağımız aritmetiğin anlamlı olması için kriptografik güvenlik açısından düzgün eğrilere ihtiyaç vardır.

Örneğin düzgün olan $y^2 = x^3 - 4x + 2$ ve $y^2 = x^3 + 2x + 1$ düzgün eliptik eğrilerinin grafiği sırasıyla aşağıdaki gibi



iken $y^2 = x^3 - 3x + 2$ tekil eliptik eğrinin grafiği ise

Şekil 13: $y^2 = x^3 - 3x + 2$ tekil eliptik eğrinin grafiği



şeklindedir.

Eliptik eğriyi sağlayan noktaları kullanarak bir ikili işlem tanımlamadan önce projektif geometriden faydalanarak bu noktalar kümesine bir nokta daha eklememiz gerekecektir. $y^2 = x^3 + ax + b$ formunda ki eliptik eğriyi uzaya yeni bir değişken ekleyerek homojen bir eşitliği sağlayacak şekilde

$$zy^2 = x^3 + axz^2 + bz^3$$

gibi düşünebiliriz. $z \neq 0$ şeklinde bu eşitliği sağlayan herhangi bir (x, y, z) üçlüsü için $(\frac{x}{z}, \frac{y}{z}, 1)$ üçlüsü de bu noktayı sağlayacaktır. Dolayısıyla bu üçlülerin izdüşümünü $y^2 = x^3 + ax + b$ denklemini sağlayan (x, y) ikililerinin denklik sınıfı gibi düşünebiliriz. Fakat $z = 0$ durumu için $(0, y, 0)$ üçlülere denklemi sağlayacaktır ve bunları düzlemde temsil edemeyeceğimizde bu noktayı eliptik eğriyi sağlayan noktalar kümesine ayrıca eklemeliyiz. Yani eliptik eğrimizi

$$E = \{(x, y) \in K \times K : y^2 = x^3 + ax + b\} \cup \{O\}$$

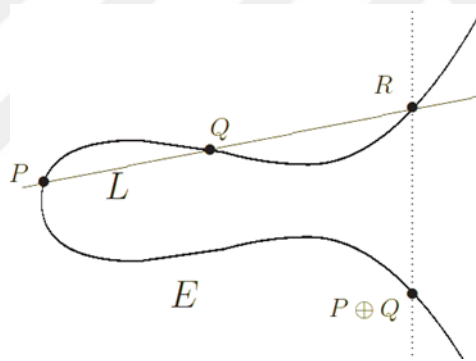
şeklinde ifade edeceğiz. Burada O yukarıda anlattığımız noktayı temsil etmektedir, bu noktaya özel olarak ilgili eliptik eğrinin sonsuz noktası denir. İleride göreceğimiz üzere tanımladığımız aritmetik için önemli bir nokta olacaktır. Dikkat edilmelidir ki Weirstrass formundaki bir eliptik eğrinin grafiği x-eksenine göre simetriktir. Ayrıca sabit bir $y \in K$ için

iki $x \in K$ değeri denklemi sağlıyor ise bu denklemi sağlayan bir başka $x \in K$ (diğer değerlere eşit olma durumuyla beraber) değeri vardır. Bu gerçek cebirin temel teoreminin sonucudur. Bu özellikler birazdan tanımlayacağımız ikili işlemin anlamlı olması adına önem arz etmektedir.

Genelliği bozmadan K 'yı reel sayılar kümesi $\mathbb{R}$ olarak düşünelim. $\oplus: E \times E \rightarrow E$ ye ikili işlemi aşağıdaki adımlarla tanımlanır:

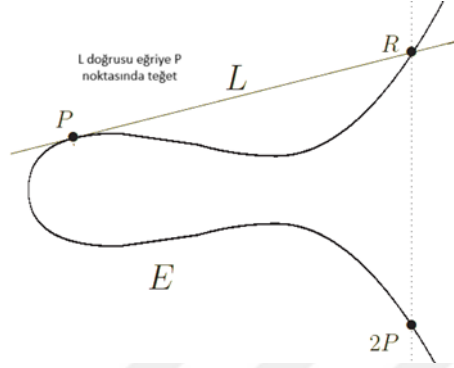
- $P \neq Q \in E$, dikey olmayan noktaları için $P \oplus Q$ değeri, ilgili eliptik eğri üzerindeki P ve Q noktalarından geçen doğrunun bu eliptik eğriyi kestiği diğer R noktasının x-eksenine göre simetriği olarak elde edilir. Bu durum aşağıdaki görsel ile gösterilmiştir.

Şekil 14: Eliptik eğri üzerinde dikey olmayan iki farklı noktanın toplamı



- $P = (x, y \neq 0)$ tipindeki bir noktanın kendisi ile toplanması, yani $P \oplus P = 2P$ değeri, de eğri üzerinde bu noktaya teğet olan teğet doğrusunun eğriyi kestiği diğer R noktasının x-eksenine göre simetriği olarak tanımlanır. Bu durum aşağıdaki görsel ile gösterilmiştir.

Şekil 15: Eliptik eğri üzerinde bir noktanın kendisiyle toplanması



- Ayrıca $P \neq Q \in E$ dikey eğrileri için $P \oplus Q = \mathcal{O}$, $P \oplus \mathcal{O} = P$ ve $P = (x, 0)$ tipindeyse $P \oplus P = 2P = \mathcal{O}$ olarak tanımlanır.

Yukarıda tasvir edilen ikili işleme göre E kümesi, birleşme, birim eleman ($\mathcal{O}$) ve ters eleman özelliklerini sağlar. Böylece bu küme bir grup yapısı teşkil eder. Daha fazla detay için [38] kaynağı incelenebilir.

Şimdi tasvir edilen bu işlemi matematiksel olarak aşağıdaki gibi ifade edelim.

1. Durum: $(x_1, y_1) = P \neq Q = (x_2, y_2) \in E$, dikey olmayan $(x_1 \neq x_2)$ noktaları $y^2 = x^3 + ax + b$ denklemini sağlasın. P ve Q noktalarından geçen doğru denklemini eğimi $m = \frac{y_2 - y_1}{x_2 - x_1}$ olmak üzere

$$y = m(x - x_1) + y_1$$

denklemini ile elde edilir. Bu doğrunun eğriyle kesişimi yanı ortak noktalarını bulmak için birbirleri yerine yazarsak

$$(m(x - x_1) + y_1)^2 = x^3 + ax + b$$

eşitliği elde edilir. Eşitlikte tüm değerleri bir tarafta toplarsak o taraf x 'in 3. dereceden bir polinomu olur yani

$$0 = x^3 - m^2 x^2 + \dots$$

şeklinde olur. Sağ taraftaki polinom aslında doğru ile eliptik eğrinin kesişim noktalarının x -değerleridir bu değerlerin ikisi x_1 ve x_2 olsun diğeri ise x_3 olsun. Bu durumda kökler toplamı gereği

$$0 = x^3 - m^2 x^2 + \dots = x^3 - (x_1 + x_2 + x_3) x^2 + \dots$$

olacağından

$$m^2 = x_1 + x_2 + x_3$$

olup

$$x_3 = m^2 - x_1 - x_2$$

olarak bulunur. Bu kesişim noktasının y -değeri ise y_3 olsun. Doğru denklemini yardımıyla ise

$$y_3 = m(x_3 - x_1) + y_1$$

olarak elde edilir. (x_3, y_3) noktasının x -eksenine göre simetriği ise $(x_3, -y_3)$ olduğundan

$$P \oplus Q = (x_3, -y_3) = (m^2 - x_1 - x_2, m(x_1 - x_3) - y_1)$$

olarak elde edilir.

2. Durum: $y^2 = x^3 + ax + b$ eğrisi üzerindeki $y_1 \neq 0$ olmak üzere $P = (x_1, y_1)$ noktası için $P \oplus P = 2P = (x_3, -y_3)$ değeri bulmak için P noktasından geçen teğet doğrusuna ihtiyacımız vardır. Biliyoruz ki teğet doğrusunun eğimi m , o noktadaki $\frac{dy}{dx}$ türev değerine eşittir. Yani kapalı fonksiyonların türevinden

$$2y \frac{dy}{dx} = 3x^2 + a$$

olduğundan eğim

$$m = \frac{3x_1^2 + a}{2y_1}$$

olarak bulunur. Benzer şekilde

$$y = m(x - x_1) + y_1$$

doğru denklemini eğride yerine koyup aynı adımları uygularsak nihai olarak x_1 çift katlı kök olacağından

$$0 = x^3 - m^2 x^2 + \dots = x^3 - (x_1 + x_1 + x_3) x^2 + \dots$$

olup

$$x_3 = m^2 - 2x_1 \text{ ve } y_3 = m(x_3 - x_1) + y_1$$

bulunur ve

$$P \oplus P = 2P = (x_3, -y_3) = (m^2 - 2x_1, m(x_1 - x_3) - y_1)$$

olarak elde edilir.

Böylece eliptik eğriler üzerindeki ikili işlemi cebirsel olarak ta nasıl elde edilebileceğini gördük. Aşağıda bu işlemler özetlenmiştir. $y^2 = x^3 + ax + b$ eğrisi için:

$$P = (x_1, y_1), Q = (x_2, y_2) \text{ öyleki } x_1 \neq x_2$$

$$P \oplus Q = (x_3, -y_3)$$

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$x_3 = m^2 - x_1 - x_2$$

$$-y_3 = m(x_1 - x_3) - y_1$$

$$P = (x_1, y_1) \text{ öyleki } y_1 \neq 0$$

$$P \oplus P = 2P = (x_3, -y_3)$$

$$m = \frac{3x_1^2 + a}{2y_1}$$

$$x_3 = m^2 - 2x_1$$

$$-y_3 = m(x_1 - x_3) - y_1$$

Böylece eğri üzerindeki işlemi basit aritmetiklerle temsil edebildik. Dikkat edilirse işlemimizde toplamsal bir gösterim kullandık. Dolayısıyla bir noktanın kendisiyle k kere toplanması

$$kP = \underbrace{P \oplus P \oplus \dots \oplus P}_{k \text{ tane}}$$

şeklinde ifade edilir. Bu ifade standart vektör uzaylarındaki skalerle çarpma işlemi ile karıştırılmamalıdır.

Buraya kadar reel sayılar cismi üzerindeki eliptik eğrilerdeki aritmetikten bahsettik. Fakat pratik uygulamalarda reel sayılar cismi kullanılmamaktadır. Bizim sonlu bir kümede

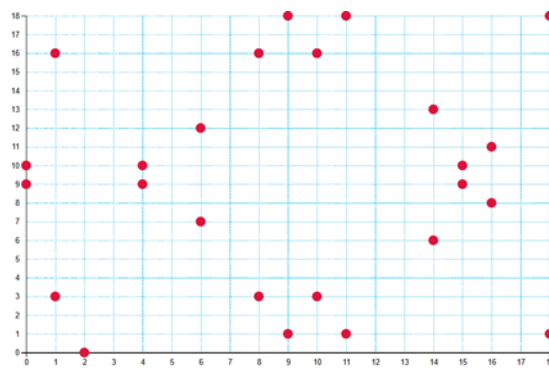
çalışmamız gerekir. Dolayısıyla sonlu cisimlere ihtiyacımız vardır. Kriptografide kullanılan eliptik eğriler ya p asal sayı olmak üzere p elemanlı sonlu cisimler ya da karakteristiği 2 olan 2^n elemanlı sonlu cisimler üzerinde tanımlanan eğriler olacaktır. Burada işlemin tanım değişmeyecek sadece bu işlemi yaparken uygulanması gereken aritmetik işlemler değişecektir. Karakteristiği 2 olan bir cisim üzerinde çalışıyorsak teğet doğrunun eğiminden bahsedemeyeceğimizden ufak bir manipölasyon yapıp eliptik eğrinin Weirstrass formu yerine $y^2 + xy = x^3 + ax + b$ tipindeki eğriler ele alınacaktır. Bu tür yapılar daha çok donanımsal tasarımlarda kullanıldığından detayları burada vermeyeceğiz.

Biz özel olarak p elemanlı $\mathbb{Z}_p$ sonlu cismini ele alalım. Bu cisimdeki aritmetiğin mod p ye göre toplama ve çarpma işlemleri olduğunu biliyoruz. $y^2 = x^3 + ax + b$ tipindeki eliptik eğrinin düzgün olması demek $\Delta = -16(4a^3 + 27b^2) \neq (0 \bmod p)$ şartının sağlanması ile denktir. Bu eliptik eğri $a, b \in \mathbb{Z}_p$ olmak üzere

$$E = \{(x, y) \in \mathbb{Z}_p \times \mathbb{Z}_p : y^2 = x^3 + ax + b\} \cup \{O\}$$

şeklinde tanımlanan kesikli noktaların kümesi olacaktır. $p = 19$ ve $y^2 = x^3 + 3x + 5$ eğrisinin $\mathbb{Z}_p \times \mathbb{Z}_p$ düzlemindeki grafiği aşağıdaki gibidir:

Şekil 16: $p = 19$ ve $y^2 = x^3 + 3x + 5$ eğrisinin $\mathbb{Z}_p \times \mathbb{Z}_p$ düzlemindeki grafiği



Görüleceği üzere simetri x-eksenine göre değil $y = 9,5$ yatay doğrusuna göredir. Diğer her şey benzer şekilde çalışmaktadır. Aşağıda, $\mathbb{Z}_p$ üzerinde tanımlanmış $y^2 = x^3 + ax + b$ eliptik eğrisi için ilgili aritmetikler özetlenmiştir:

$P = (x_1, y_1), Q = (x_2, y_2)$ öyleki $x_1 \neq x_2$

$$P \oplus Q = (x_3, -y_3)$$

$$m = (y_2 - y_1)(x_2 - x_1)^{-1} \pmod{p}$$

$$x_3 = m^2 - x_1 - x_2 \pmod{p}$$

$$-y_3 = m(x_1 - x_3) - y_1 \pmod{p}$$

$P = (x_1, y_1)$ öyleki $y_1 \neq 0$

$$P \oplus P = 2P = (x_3, -y_3)$$

$$m = (3x_1^2 + a)(2y_1)^{-1} \pmod{p}$$

$$x_3 = m^2 - 2x_1 \pmod{p}$$

$$-y_3 = m(x_1 - x_3) - y_1 \pmod{p}$$

Buradaki üssün (-1) olması o değerin modüler p ye göre çarpımsal tersini göstermektedir.

Eliptik eğrilerde ayrık logaritma problemi: $\mathbb{Z}_p$ çarpımsal grubunda ifade ettiğimiz ayrık logaritma problemini $\mathbb{Z}_p$ üzerinde tanımlanmış bir E eliptik eğrisi içinde ifade edebiliriz. Burada dikkat edilmesi gereken durum gösterimlerin toplamsal olarak kullanıldığı gerçeğidir. Verilen bir E eliptik eğrisi ve $P \in E$ noktası için

$$kP = Q$$

şartını sağlayan Q noktası biliniyorken k tamsayı değerini belirlemek problemine eliptik eğrilerdeki ayrık logaritma problemi denir. Bu problem yeterince çok nokta içeren eğriler için zor bir problem olarak bilinmektedir.

Şimdi eliptik eğrilerde tanımladığımız bu aritmetik yardımıyla eliptik eğri kriptosistemini verebiliriz.

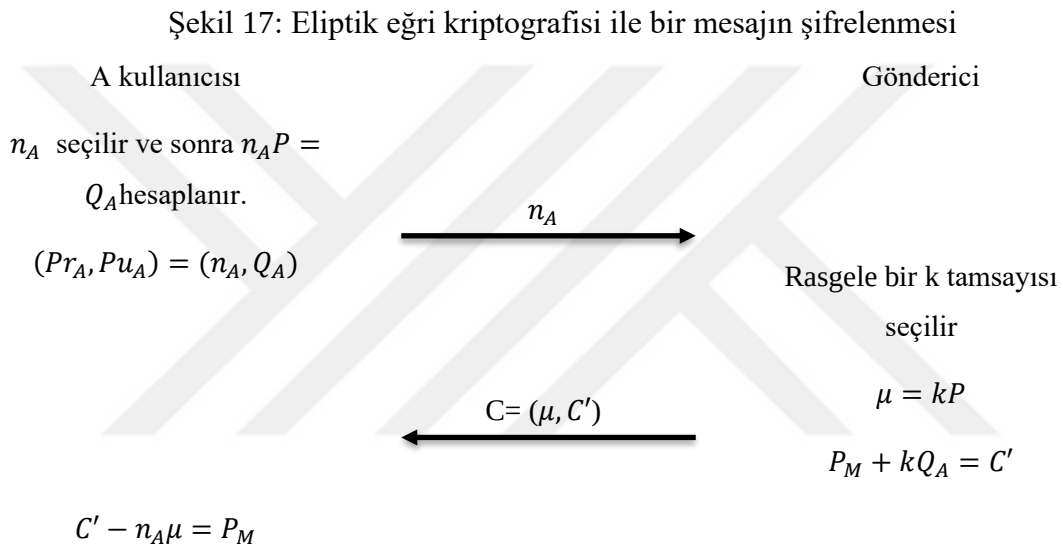
- **Genel parametrelerin oluşturulması:** Asimetrik kriptografi sistemine dahil olan her kullanıcının kullanacağı parametrelerin belirlenmesidir. Uygun bir p asal sayısı ve $\mathbb{Z}_p$ üzerinde bir eliptik eğri belirlenir ve bu grupta mertebesi yeterince yüksek bir P noktası belirlenir.
- **Anahtar üretimi:** Sistemdeki bir A kullanıcısı kendi açık ve gizli anahtarını belirler. Gizli anahtar olarak bir n_A tamsayısı seçildikten sonra $n_AP = Q_A$ olacak şekildeki Q_A noktası açık anahtar olarak belirlenir.
- **Şifreleme:** Bir M mesajı eğri üzerindeki bir P_M noktasına gelecek şekilde kodlanır. Daha sonra tek kullanımlık bir k tamsayısı seçilir. A kullanıcısına mesaj iletilmek istendiği durumda şifreli mesaj $C = \{kP, P_M + kQ_A\} = \{\mu, C'\}$ eliptik eğri üzerindeki noktaları olarak elde edilir.

- **Deşifreleme:** $C = \{kP, P_M + kQ_A\} = \{\mu, C'\}$ alan A kullanıcısı,

$$C' - n_A \mu = P_M + kQ_A - n_A kP = P_M + kQ_A - kQ_A = P_M$$

işlemi ile P_M yi bulup kodlamayı tersine yapıp M mesajını elde eder.

Yukarıda anlatılan adımlar aşağıdaki şemada özetlenmiştir. Önceden belirlenmiş $\mathbb{Z}_p$ cismi ve onun üzerindeki bir eliptik eğri ile bu eliptik eğri grubunda mertebesi yeterince yüksek bir P noktası için, belirlenmiş bir P_M mesajı şifreleme şeması:



Eliptik eğri sistemleri RSA sisteminde 2048 bit ile sağlanacak güvenlik seviyesini, 224-255 bitlik parametreler ile sağlar. Aynı zamanda şifreli mesaj boyuda kısılacaktır ve band genişliği dar olan iletişim ağlarında avantaj oluşturacaktır.

Tabi ki bu avantajlarının yanında çeşitli dezavantajları da mevcuttur. Bunlardan önemlileri aşağıda listelenmiştir.

- Uygun bir p asal sayısının ve yeterince nokta içeren uygun bir eliptik eğrinin belirlenmesi. Bunun için eliptik eğri veri tabanları oluşturulmuştur.
- Bilinen bir eliptik eğri üzerinde mertebesi yeterince yüksek bir P noktasının belirlenmesi.

- Herhangi bir mesajın eğri üzerinde bir noktaya denk gelecek şekilde kodlanması problemi.

3.3.5.4. Dijital İmza Algoritmaları (DSA ve ECDSA)

Daha önceden belirttiğimiz üzere herhangi bir asimetrik şifreleme mekanizması bir imzalama algoritması olarak kullanılabilir. Dolayısıyla RSA algoritması ve bir önceki bölümde anlattığımız eliptik eğri tabanlı şifreleme algoritması aynı zamanda dijital imzalama algoritmaları olarak kullanılabilir. Fakat verimlilik ve büyük imza boyutları yüzünden sadece imzalamaya yönelik algoritmalara ihtiyaç doğmuştur. Bu ihtiyaca yönelik 1991 yılında standartlaşmış DSA (Digital Signature Algorithm) [39] geliştirilmiş. DSA'daki güvenlik seviyesi daha küçük parametrelerle gerçekleyen ECDSA (Elliptic Curve Digital Signature Algorithm) 1992 yılında sunulmuş ve 2000 yılında standartlaşmıştır [40]. Bu bölümde bu iki algoritmanın detayları anlatılacaktır. Daha fazla detay ve teorik ilişki için [33] kaynakları incelenebilir.

DSA ile başlayalım. DSA yeterince büyük bir p asal sayısı için $\mathbb{Z}_p^*$ çarpımsal grubu üzerindeki ayrık logaritma probleminin zorluğuna dayanmaktadır. Aşağıda en düşük güvenlik seviyesindeki DSA parametreleri için algoritmanın adımları listelenmiştir. Genel parametreler daha sonra özetlenecektir.

- **Genel parametreler:** $512 \leq L \leq 1024$ ve L , 64'ün bir katı olmak üzere L -bit bir p asal sayısı belirlenir. $(p - 1)$ sayısını bölen 160 bit uzunluğunda bir q asal sayısı seçilir. $2 \leq h \leq p - 2$ olacak şekilde rasgele bir h değeri seçilir (genel kullanımda $h = 2$ seçilir) ve $g = h^{(p-1)/q} \pmod{p}$ değeri belirlenir. Eğer bu değer 1 ise yeniden h değeri seçilip 1 olmayacak şekilde ayarlanır. Daha sonra 160 bit çıktı veren uygun bir H özetleme fonksiyonu belirlenir. Bu parametreler herkes tarafından bilinen açık bilgilerdir.
- **Anahtar üretimi:** Bir kullanıcı $1 \leq x \leq q - 1$ olacak şekilde bir gizli anahtar x değeri belirler ve bu x değeri ile $y = g^x \pmod{p}$ açık anahtarını hesaplar.
- **İmzalama:** Bir kullanıcı bir M mesajını imzalamak için öncelikler rasgele bir $1 \leq k \leq q - 1$ olacak şekilde k değeri belirler. Bu k değeri her farklı mesajda değiştirilmelidir. Daha sonra

$$r = (g^k \pmod{p}) \pmod{q} \quad \text{ve} \quad s = k^{-1}(H(M) + xr) \pmod{q}$$

değerleri hesaplanarak imza çifti (r, s) oluşturulur. Eğer $s = 0$ olarak hesaplanmış ise başa dönülüp yeni bir k değeri belirlenir.

- **Doğrulama:** (r, s) imza çiftini alan bir kişi imzayı atan kişinin açık anahtarı y yardımıyla

$$u_1 = H(M)s^{-1} \pmod{p}, \quad u_2 = rs^{-1} \pmod{p}$$

$$v = (g^{u_1}y^{u_2} \pmod{p}) \pmod{q}$$

değerlerini hesaplar ve $v = r$ eşitliğini test eder. Eğer eşitlik sağlanırsa imza geçerlidir sağlanmaz ise imza reddedilir.

Şimdi bu eşitlik sağlandığında imzanın geçerli olma nedenini görelim.

$$s = k^{-1}(H(M) + xr) \pmod{q}$$

olduğundan açık bir şekilde

$$k = s^{-1}(H(M) + xr) \pmod{q}$$

olacaktır. Buradan

$$\begin{aligned} v &= (g^{u_1}y^{u_2} \pmod{p}) \pmod{q} = (g^{H(M)s^{-1}}y^{rs^{-1}} \pmod{p}) \pmod{q} \\ &= (g^{H(M)s^{-1}}g^{xrs^{-1}} \pmod{p}) \pmod{q} \\ &= (g^{s^{-1}(H(M)+xr)} \pmod{p}) \pmod{q} = (g^k \pmod{p}) \pmod{q} = r \end{aligned}$$

olduğu görülebilir. Aşağıdaki şema ile algoritma özetlenmiştir.

Şekil 18 : Dijital imza algoritması (DSA)

Güvenlik seviyelerine göre uygun bit uzunluklarında p ve q asal sayıları ve q nun bit uzunluğu kadar çıktı veren H özetleme fonksiyonu belirlenir.

$$2 \leq h \leq p - 2 \quad \text{için} \quad g = h^{(p-1)/q} \pmod{p} \quad \text{belirlenir.}$$

İmzalayan

Doğrulayan

$$1 \leq x \leq q - 1 \quad \text{olacak şekilde}$$

$$\begin{array}{ccc}
y = g^x \pmod{p} & & \\
(Pr_A, Pu_A) = (x, y) & \xrightarrow{y} & \\
1 \leq k \leq q - 1 \text{ için} & & u_1 = H(M)s^{-1} \pmod{p} \\
r = (g^k \pmod{p}) \pmod{q} & & u_2 = rs^{-1} \pmod{p} \\
s = k^{-1}(H(M) + xr) \pmod{q} & \xrightarrow{(r,s)} & v = (g^{u_1}y^{u_2} \pmod{p}) \pmod{q} \\
& & v = r (?)
\end{array}$$

DSA algoritmasının daha yüksek güvenlik seviyesindeki parametreleri, p ve q asal sayılarının bit uzunluklarının (2048,224), (2048,256) ve (3072,256) olarak seçilmektedir. Daha detaylı bilgi için [41] kaynağı incelenebilir.

Şimdi ECDSA algoritmasına geçebiliriz. Benzer bir yapının eliptik eğriler üzerindeki aritmetiklerle inşa edilmesiyle elde edilen algoritmadır. Aynı güvenlik seviyesinde daha az hesaplama ve düşük imza boyutu sunması nedeniyle düşük donanıma sahip yerlerde tercih edilmektedir. Günümüzde birçok blok zinciri teknolojisi dijital imzalama algoritması olarak ECDSA'yı tercih etmektedir.

Şimdi algoritmanın adımlarını verelim.

- **Genel parametreler:** Uygun bir p asal sayısı için $\mathbb{Z}_p$ cismi üzerinde bir E eliptik eğrisi belirlenir. Bu eğrinin nokta sayısını bölen bir yeterince büyük q asal sayısı belirlenir ve mertebesi q olan $P \in E$ noktası belirlenir.
- **Anahtar üretimi:** Bir kullanıcı $2 \leq d \leq q - 1$ olacak şekilde gizli anahtar belirler ve bu anahtar yardımıyla $dP = Q$ olacak şekilde açık anahtarı belirler.
- **İmzalama:** Bir kullanıcı bir M mesajını imzalamak için öncelikle $1 \leq k \leq q - 1$ olacak şekilde rasgele bir k değeri belirler. Bu k değeri her farklı mesajda değiştirilmelidir. Daha sonra,

$$kP = (x_1, y_1) \text{ olacak şekilde } r = x_1 \pmod{q}$$

$$s = k^{-1}(H(M) + dr) \pmod{q}$$

değerleri hesaplanarak imza çifti (r, s) oluşturulur. Eğer $s = 0$ olarak hesaplanmış ise başa dönülüp yeni bir k değeri belirlenir.

- **Doğrulama:** (r, s) imza çiftini alan bir kişi imzayı atan kişinin açık anahtarı Q yardımıyla

$$u_1 = H(M)s^{-1}(\text{mod } q), \quad u_2 = rs^{-1}(\text{mod } q)$$

$$u_1P + u_2Q = (x_0, y_0) \text{ olacak şekilde } v = x_0 (\text{mod } q)$$

değerlerini hesaplar ve $v = r$ eşitliğini test eder. Eğer eşitlik sağlanırsa imza geçerlidir sağlanmaz ise imza reddedilir.

DSA algoritmasındakine benzer şekilde doğrulama adımını kontrol edersek,

$$\begin{aligned} u_1P + u_2Q &= H(M)s^{-1}P + rs^{-1}Q = H(M)s^{-1}P + drs^{-1}P = (H(M)s^{-1} + drs^{-1})P \\ &= kP = (x_1, y_1) \end{aligned}$$

olduğundan $v = r$ ile imzanın geçerliliği kontrol edilir.

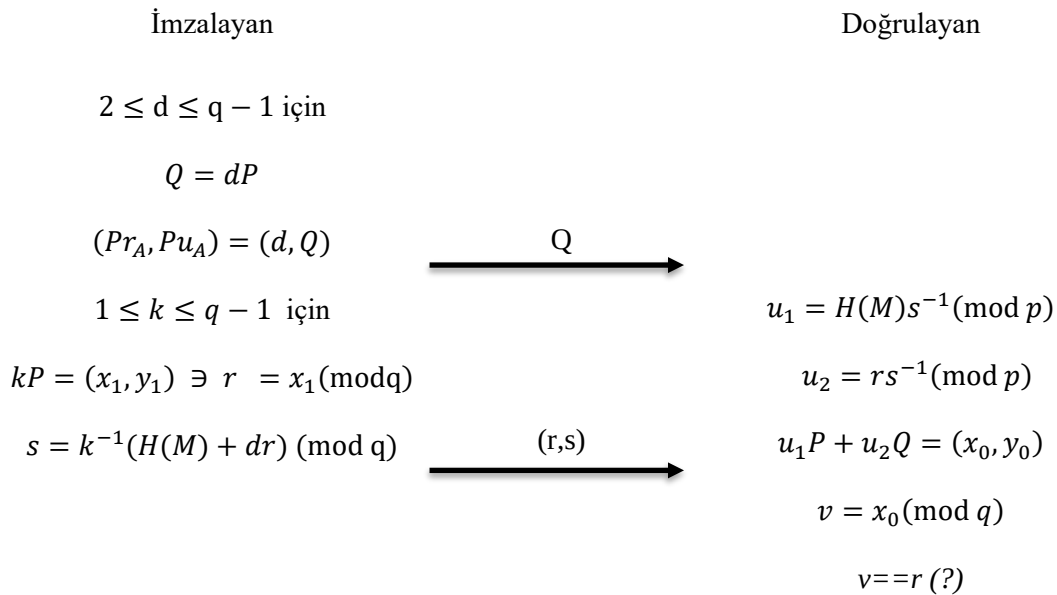
ECDSA için anlatılan bu adımlar aşağıdaki şema ile özetlenebilir.

Şekil 19: Eliptik eğri tabanlı dijital imzalama algoritması (ECDSA)

Güvenlik seviyelerine göre uygun bit uzunluklarında p asal sayısı için $\mathbb{Z}_p$ cismi belirlenir.

Uygun bir H özetleme fonksiyonu belirlenir.

$\mathbb{Z}_p$ üzerinde uygun bir E eliptik eğrisi ve bu eğride mertebesi en büyük asal sayı q olan P noktası seçilir



En yaygın olarak kullanılan blok zinciri teknolojileri Bitcoin ve Ethereum dijital imzalama algoritması olarak ECDSA yı kullanmaktadır. Her iki teknoloji de secp256k1 koduyla betimlenmiş eliptik eğri kullanılmaktadır. Bu eğri

$$p = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$$

asal sayısı için $\mathbb{Z}_p$ cismi üzerindeki

$$y^2 = x^3 + 7$$

formu ile tanımlanan eliptik eğridir. Buradaki P noktasının x-bileşeni

550662630222773436695787188951685343262506034537775941755001873603891 6729240

sayısı y-bileşeni ise

32670510020758816978083085130507043184471273380659243275938904335757337482424

sayısı olarak belirlenmiştir.

3.3.6. Asimetrik Kriptografinin Geleceği

Blok zinciri teknolojileri gerek adres üretiminde gerekse dijital imzalamada büyük ölçüde asimetrik kriptografiyi kullanmaktadır. Hatta daha ileri uygulamalarda daha kapsamlı mekanizmalar araştırılmakta ya da inşa edilmektedir. Fakat günümüzde kullanılan çoğu asimetrik kriptografi mekanizmaları önemli bir tehdit altındadır. Quantum hesaplama yapabilen cihaz teknolojilerin hızla gelişmesi asimetrik kriptografiyi önemli ölçüde tehdit etmektedir. Çünkü bazı bilinen kuantum algoritmalar [42] yardımıyla çarpanlara ayırma, ayrık logaritma vs. gibi problemler kuantum hesaplamalar ile pratik bir şekilde çözülebilmektedir. Dolayısıyla çok yakın olmasa da yakın bir gelecekte bu güce sahip cihazların kullanılıyor olması sayesinde önceki bölümlerde verdiğimiz asimetrik kriptografi algoritmalarının güvenlik seviyeleri oldukça düşecektir. Bu duruma karşı günümüzde yeni asimetrik şifreleme ve dijital imzalama algoritmaları tasarlanmaya başlanmış hatta bunlardan bazıları standart olarak belirlenmiştir [43].

Anlaşılabacağı üzere birçok blok zinciri teknolojisi de bu tehdit ile karşı karşıyadır. Dolayısıyla blok zinciri alanında kuantum cihazlara dayanıklı yapıların uygulanabilirliği alanlarında

arařtırmalar yoęunlařmıř durumdadır ve biręok teknoloji gnmz haliyle kullanılamaz hale gelecektir.

3.4. Protokoller

Kriptografi bilimindeki nemli alt bařlıklardan biri de protokollerdir. Geręek hayatta ihtiyaę duyulan bazı durumların dijital ortamlarda gvenli bir řekilde yapılması protokoller yardımıyla olur. Kriptografi protokolleri iki veya daha fazla muhatap arasında geęen algoritmalar btn olarak tanımlanabilir. Genellikle interaktif olarak geręekleřmektedir. Yani iki veya daha fazla taraf birbirleriyle iletiřim kurarlar ve iletiřim sonu oluřan parametreler gereęi algoritmalar ęalıřır. Fakat ęeřitli araęlar yardımıyla interaktif olmayacak řekilde de tasarlanabilirler.

Protokollere en temel rnek olarak Blm 3.3.5.1 bahsettięimiz Diffie-Hellman anahtar anlařma protokol verilebilir. Bu protokol iki kiři tarafından gizli bilgileri aęıęa ıkarmadan simetrik řifrelemede ortak bir anahtarı hesaplamaya yararmaktadır.

Blok zinciri teknolojileri basit transferler olmanın dıřında reel hayatın bize sunduklarını dijital ortamda sunmak adına yeniliklere ihtiyaę duymaktadır. Bu yenilikleri de gvenli bir řekilde bu ortamlarda geręekleyebilmek adına kriptografik protokoller kullanılmaktadır. Bu blmde bazı protokollerden genel hatları ile bahsedilerek bunları kullanan blok zinciri teknolojilerine rnekler verilecektir.

3.4.1. Taahht Protokol

Taahht protokol (commitment scheme), bir kullanıcının dięerine bir taahhtte bulunarak ileri bir tarihte geręekleřmek zere bir bilginin saklanmış detayını gndermesidir. Bu ařamaya taahht etme (commit) ařaması adı verilir. Alıcı ise bu aldıęı veriyi ileride kullanmak zere saklar. Zamanı geldięinde gnderici commit ařamasında veriyi saklarken kullandıęı bilgileri alıcıya gnderir ve alıcı aynı saklama iřlemi sonunda daha nce depoladıęı veriyi elde edip edilemeyeceęini kontrol eder. Bu ařamaya da aęıęa ıkarma (reveal) ařaması denir.

Birçok farklı varyasyonlarda tanımlanmış bu tür de protokol mevcuttur [44]. İleriye dönük reel hayat gerçeklerini blok zinciri teknolojilerinde gerçekleyebilmek adına bu tür protokollere ihtiyaç vardır. Monero blok zinciri [45] (pedersen commitment) teknolojisi bu protokoller yardımıyla bir tasarım gerçekleştirmiştir.

3.4.2. Kimlik Doğrulama Protokolleri

Reel hayatta güvenli bir yere girerken kimliğimizi fiziksel olarak ibraz ederiz ve o kişinin kendimiz olduğumuzu ispatlarız veya parmak izi vb. fiziksel kanıtlar sunarak kimliğimizi doğrularız. Dijital ortamlarda kimlik doğrulamak (authentication) bu kadar pratik değildir. Çünkü sizin siz olduğunuzun kontrolünü bir algoritma gerçekleştirecektir. Kimlik doğrulama protokolleri bu problemin çözümü için sunulan çözümlerdir. Genel olarak iki aşamadan oluşmaktadır. Kayıt ve doğrulama aşamaları. Öncelikle kullanıcı ortama kayıt olur ve sonra o ortama her girişinde kimlik doğrulamasını gerçekleştirir. En yaygın kullanılan yapısı şifre tabanlı protokollerdir. Fakat çeşitli yapılarda kimlik doğrulama protokolleri vardır [46].

Blok zinciri teknolojilerinde ise kullanıcı dostu cüzdan uygulamaları hesabınıza erişmek ve sizin kimlik doğrulamanızı gerçekleştirmek için bu protokollerden faydalanmaktadır. Dolayısıyla cüzdan uygulaması kullanmadan önce o uygulamanın bu işlemleri nasıl yapıldığını bilmek kullanıcıya güven vermektedir. Ayrıca literatürde kimlik doğrulamalarının blok zinciri teknolojisini araç olarak kullanılarak yapılması adına önerilen birçok çalışma mevcuttur.

3.4.3. Sıfır Bilgi İspatları

Sıfır bilgi ispatı (zero-knowledge proof), bir protokol olarak değerlendirilebilecek kriptografik bir tekniktir. Bu protokolün temel amacı, bir tarafın (kanıtlayıcı) belirli bir bilgiye sahip olduğunu, karşı tarafa (doğrulamacı) söz konusu bilginin kendisini ifşa etmeden ispatlamasıdır. Süreç, dijital ortamda bilgi paylaşmadan doğrulama yapmayı mümkün kılar. Kavramsal olarak, bu durum bir sınav örneğiyle açıklanabilir: Bir öğrenciye yöneltilen matematik sorusu, onun bilgi düzeyini test eden bir zorluk işlevi görür. Öğrencinin bu soruya

doğru yanıt vermesi, sorunun çözümüne hâkim olduğunu gösterir; ancak yalnızca sonucu paylaşması, çözüm yöntemini açıklamadan bilgiyi ispatlaması anlamına gelir. Sıfır bilgi ispatlarında da benzer şekilde, bilgiye sahip olunduğu ispatlanır, fakat bilginin kendisi asla açığa çıkmaz.

Teknik olarak tanımlamak gerekirse, bir $f(x) = y$ hesaplamasında y değeri herkes tarafından bilinirken x değerinin bildiğini ispat etmeye çalışmak bir bilgi protokolünün amacı olarak tanımlanabilir. En basiti x bilgisini karşı tarafa verip hesaplamayı yapmasını sağlamak örneklenebilir. Önceki bölümlerde bahsettiğimiz dijital imza algoritmaları o gizli bilgiye sahip olduğunuzu ilgili mesaj ile karşı tarafa ispat ettiğiniz bir mekanizma olarak düşünülebilir.

Anlaşılabileceği üzere bir ispat protokolünde iki taraf vardır. Taraflardan biri ispat eden diğeri ise ispatı doğrulayan olarak adlandırılır. Bir bilgi ispatı protokolünün sağlaması gereken özellikler aşağıda verilmiştir.

- **Tamlık (Completeness):** Eğer ispat edenin iddiası doğruysa, ispat eden, doğrulayan kişiyi ihmal edilebilir bir olasılık dışında ikna etmelidir. Eğer %100 ikna edebiliyorsa o ispata özel olarak “mükemmel tamlık” özelliğini sağlıyor denir.
- **Dayanım (Soundness):** Eğer ispat eden hileci ise doğrulayanı sadece ihmal edilebilir bir olasılık dahilinde elindeki sahte bilgiye ikna edebilir. Onun dışında doğrulayan onun yanlış bilgi verdiğini fark edebilmelidir.

Bu iki özelliğin yanında;

- **Sıfır bilgi (Zero-Knowledge):** Doğrulayanın, protokol sonunda ispat edenin iddiasının doğruluğu dışında başka bir bilgi elde etmemesi,

durumu da eklenirse ilgili bilgi ispatı protokolü sıfır bilgi ispatı protokolü olarak adlandırılır. Çeşitli sıfır bilgi ispatı protokolleri için [47] incelenebilir.

Blok zinciri teknolojilerinde mahremiyeti sağlamak son zamanlarda en önemli problemlerden biri olarak görülmektedir. Blok zincirleri doğası gereği şeffaf olduğundan tüm veriler okunabilir durumdadır. Verilerin gizli ama doğru bir şekilde tutulduğunu

garantilemek sıfır bilgi ispatı protokolleri sayesinde gerçekleşmektedir. Z-cash teknolojisi zerocash protokolünü [48], Mina teknolojisi ZK-SNARK protokolünü [49] kullanan popüler blok zinciri teknolojilerine örnek verilebilir. ZK-STARK protokolü [50] ise blok zinciri teknolojisinde daha hızlı ve ölçeklenebilir sıfır bilgi protokolü sunarken günümüzde henüz adapte edilmiş bir blok zinciri projesi mevcut değildir.

3.4.4. Eşik (Threshold) Kriptografi

Bazı durumlarda gizli bir bilginin birden çok tarafa dağıtılması gerekebilir ve bu gizli bilgi yalnızca belli sayıda kişinin bir araya gelmesi ile açığa çıkar. Örnek olarak bir bankadaki kişisel kasayı açmak için, biri kasa sahibinde, diğeri ise bir banka çalışanında bulunan iki anahtarın kullanılması durumu verilebilir. Bu tür yapılar dijital ortamların kullanımının daha interaktif hale gelmesini kolaylaştırdığı gibi güvenliği de arttırmaktadır. Sır paylaşım şemaları, anahtar dağıtma protokolleri, halka imzaları, çoklu imzalar vs. gibi protokoller bu kriptografik yapıya örnek olarak verilebilir.

Blok zinciri teknolojisinde ortak hesap kullanımı gibi bazı fonksiyonel özellikleri sunabilmek için eşik kriptografi araçlarına ihtiyaç duyulmaktadır. Ayrıca ileride blok zinciri yapılarının farklı alanlarda uygulamalarında örneğin e-oylama vs. gibi eşik kriptografi araçları önemli rol oynayacaktır. Bu yapılar genelde karmaşık matematiksel ilişkileri kullandıkları için blok zinciri doğasına çok uygun gözükmemektedir. Dolayısıyla son zamanlardaki çalışmalar bu yapıların pratik olarak blok zinciri teknolojilerine adaptasyonları üzerine yoğunlaşmıştır.

Kripto varlıkların saklandığı cüzdanlarda özel anahtarlar genelde tek bir kişidedir. Eşik sistemleri sayesinde özel anahtarlar parçalara bölünür ve birden fazla kişide tutulur. Bu yapı, kurumsal cüzdan güvenliği için kritik önemdedir. Örnek: Fireblocks, ZenGo gibi cüzdanlar threshold imza altyapısı kullanır.

3.4.5. Güvenli Çok Taraflı Hesaplama(Secure Multi-Party Computation, SMPC)

Her bir kullanıcının kendi gizli bilgisini açığa çıkarmadan o bilgiyi kullanarak diğer kullanıcılarla beraber ortak hesaplama yapmasını sağlayan mekanizmalardır. Örnek Diffie-

Hellman anahtar anlařma protokolü verilebilir. Her iki tarafta gizli anahtarlarını açık etmeden ortak bir simetrik řifreleme anahtarını hesaplıyorlar. Bu tür yapılar dijital ortamlarda, reel hayat uygulamalarını gerçeklemek için sıklıkla kullanılmaktadır. Bilinen birçok e-oylama sistemi özelinde bu protokolleri kullanmaktadır. Homomorfik řifreleme yapıları da bu protokolün önemli örneklerindendir.

Blok zinciri teknolojisi üzerinde bu tür homomorfik řifreleme yöntemlerinin uygulamaları çalışılmaktadır. Ayrıca blok zinciri teknolojileri yardımıyla güvenli çok taraflı hesaplamaların nasıl yapılabileceğine dair arařtırmalar yapılmaktadır.

3.5. Blok Zincirinde Kriptografi

Blok zinciri bazı kriptografik temelleri bir araya getirerek deęiřtirilemez ve merkezi bir otorite tarafından tutulmayan daęıtık bir veri yapısı olarak görülebilir. Daha önceleri bazı dijital para girişimleri yapılmıř fakat çift harcama gibi bir probleme çözüm üretememiřlerdir. Ayrıca her bir girişim ticari bir kuruluş tarafından yapıldığı için güven problemi de olmuřtur. Bunlardan en önemlisi David Chaum tarafından yeni bir dijital imza yapısı ile inřa edilmiř “digital cash” dijital varlığı gösterilebilir [4].

2008 yılında yayınlanan Bitcoin whitepaper’ında [9], Satoshi Nakamoto tarafından daha önceki bu girişimlerin çözemediğı problemlere önerilen çözümler ve veri yapısı türü blok zinciri teknolojisinin doęuřu olmuřtur. O çalışmada blokların zinciri olarak tasvir edilen yapı ilerleyen yıllarda günümüzde kullandığımız şekliyle “blok zinciri” olarak literatürde yer almıřtır. Bu bölümde blok zinciri teknolojisinin matematiksel arka planına bakmaya çalışacağız. Bölüm 4’te Bitcoin özelinde detaylı olarak yapılardan bahsedilirken, bu bölümde biraz daha teknik detaylardan bahsedilecektir. Bu bölümde anlatılacaklar genel olarak sistemi özetlerken detaylar farklı zincir teknolojilerinde deęiřiklik göstermektedir.

3.5.1. Blok Zinciri Bileşenleri

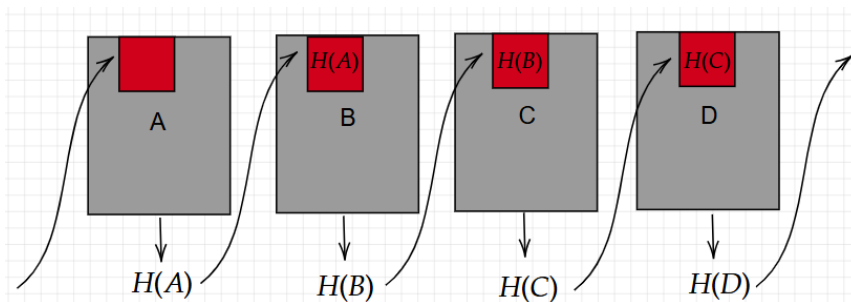
Blok zinciri özelinde bir dağıtık veri yapısı olarak ele alınmalıdır. Dolayısıyla çeşitli verileri tutması gerekmektedir. Bu verileri tutan yapılara bloklar denir. Fakat veriler tek bir blokta değil birbirine bir şekilde bağlanmış ardışık bloklar bütününde tutulmaktadır. Zincir kavramı da buradan gelmektedir. Verinin ise bir sahibi olmadığından dağıtık yani tüm kullanıcılar tarafından tutulması gerekmektedir. Özetlersek bir blok zincirindeki bileşenler bloklar, bloklar içerisindeki veriler ve kullanıcılar olarak sayılabilir.

Blok zinciri teknolojisinde en önemli bileşen bloklardır ve bu blokların nasıl değiştirilemeden tüm kullanıcılar tarafından saklandığı kriptografinin amaçlarından bütünlük için verilebilecek en güzel örneklerden biridir.

3.5.1.1. Bloklar

Bir blok zincirinde verileri depolayan en küçük veri yapısı olarak düşünülebilir. Blokların bütünü ise blok zincirini oluşturmaktadır. Bir blok içerisinde sistemin verileri dışında belli başlı veriler tutulmaktadır. Genel olarak bir blok işlemleri (transaction), zaman damgası, bir önceki bloğun özet değeri, PoW değeri vs. çeşitli veriler bulunmaktadır. Buradaki en önemli detay bir önceki bloğun özet değeri olarak düşünülebilir. Çünkü o blokta bir öncekini barındıracağından aslında bir blokta kendinden önceki tüm blokların özetlerinin izi bulunulacaktır. İşte bu durumda blok zincirinin değiştirilemez olmasını gerektirir. Çünkü önceki verilerin birinde yapılmaya çalışılacak bir değişiklik sonraki tüm blokları etkileyecektir veya etkilemeyecek bir değişiklik özetleme fonksiyonlarının ikincil ön görüntü direnci gereği pratik olmayacaktır. Bu durumu aşağıdaki görsel ile özetleyebiliriz.

Şekil 20: Blok hash değerlerinin bir sonraki blokta yer alması



Görüleceği üzere her bir blok aslında kendisinden önceki tüm blokların parmak izini barındırmaktadır. Bu yapı sayesinde blok zinciri bütünü oluşur. Dolayısıyla kriptografik güvenli bir özetleme fonksiyonu sayesinde blok zincirindeki tüm verilerin bütünlüğü korunur ve değiştirilemez bir yapıya sahip olması sağlanır. Ayrıca bir blok zincirinin oluşabilmesi için başlangıç bloğuna ihtiyaç vardır. Bu blok genesis blok olarak adlandırılır. Blok yapıları genelde bu şekildeyken blok zinciri teknolojisine göre farklılıklar göstermektedir. Aşağıda Bitcoin zincirinin 889,503. bloğunun içeriğinin özet görseli verilmiştir [51].

Şekil 21: Bitcoin zincirinin 889,503. bloğu

Details			
Hash	00000-6c514	Depth	3
Capacity	145.22%	Size	1,522,709
Distance	22m 22s	Version	0x2d732000
BTC	6,682.2948	Merkle Root	db-39
Value	\$588,491,397	Difficulty	113,757,508,810,854.00
Value Today	\$588,144,920	Nonce	3,496,797,487
Average Value	2.2476605616 BTC	Bits	386,038,124
Median Value	0.00329054 BTC	Weight	3,993,086 WU
Input Value	6,682.34 BTC	Minted	3.13 BTC
Output Value	6,685.47 BTC	Reward	3.17331657 BTC
Transactions	2,973	Mined on	26 Mar 2025 11:40:57
Witness Tx's	2,673	Height	889,503
Inputs	7,722	Confirmations	3
Outputs	8,509	Fee Range	1-96 sat/vByte
Fees	0.04831657 BTC	Average Fee	0.00001625
Fees Kb	0.0000317 BTC	Median Fee	0.00000679
Fees kWU	0.0000121 BTC	Miner	Mara Pool

Buradaki kavramların önemlileri ilerleyen bölümlerde bahsedilecektir. Görüldüğü üzere bu blokta aynı zamanda 2.973 adet işlem kayıtlıdır.

3.5.1.2. İşlemler

İşlemler blok zinciri teknolojisinin amacına hizmet eden verilerdir. Bu veriler değiştirilemez olarak bloklarda tutulur ve blok zinciri bütünü tutan her bir kullanıcı tarafından izlenebilir. Bitcoin özelinde bu işlemler BTC transferleri olarak söylenebilir. Fakat Ethereum daki veriler sadece varlık transferleri değildir aynı zamanda akıllı kontratlar da

bloklarda tutulan verilerdir. Blok zinciri teknolojisinin çıkış amacı olan dijital varlık transferi özelindeki işlemlere değinmek uygun olacaktır. İlerleyen bölümde bu yapılardan daha detaylı bahsedilecektir. Ama burada blok zinciri teknolojilerinde çift harcamanın önüne geçen önemli bir keşif vardır. Bir işlem ilgili dijital varlık oluştuktan sonra yapılır ve o varlık için yapılan tüm sonraki transferler önceki transferin bilgileri ile beraber işlenir. Yani bir varlık ağaç yapısı şeklinde dallanarak ağdaki kullanıcılara dağılır. Bu da aynı varlığın birden fazla kere işleme tabi tutulmasını engeller. Bu her bir işleme patent sertifikası konması anlamına gelir. Bu mantık fiziki işlemlerdeki çek ile benzerlik göstermektedir.

3.5.1.3. Kullanıcılar

Bir blok zinciri teknolojisinin sağlıklı çalışması ve canlı kalması için o teknolojinin kullanıcıları olması gerekmektedir. Farklı tipte kullanıcılar olacağından her kullanıcının rolü aynı değildir. Kullanıcı rolleri genel olarak aşağıdaki gibi verilebilir:

- **Yalın kullanıcılar:** Bu kullanıcılar blok zinciri teknolojisinin sunduğu hizmetten faydalanan kullanıcılardır. Genellikle bir yardımcı araç cüzdan ile sisteme dahil olarak sadece işlem gerçekleştiren kullanıcılardır.
- **Madenciler – Hissedarlar:** Blok zinciri teknolojilerinin olmazsa olmaz fikir birliği algoritmalarına katkıda bulunan kullanıcılardır. PoW kullanan zincirlerde madenci olarak adlandırılırlar. PoS ve benzeri yapıları kullanan zincirlerde ise hissedarlar olarak adlandırılırlar.
- **Uç birimler:** Blok zinciri verisinin tamamını (tam uç birim) veya bir kısmını tutan (hafif uç birim) kullanıcılara verilen addır. Bu kişiler ayrıca gönüllü olarak doğrulayıcı rolünü de üstlenebilirler. Doğrulayıcılar eklenen blokların doğruluklarını kontrol ederler.

3.5.2. Dijital İmza Algoritmaları ve Blok zinciri

Özetleme fonksiyonlarıyla beraber blok zinciri teknolojilerinin hepsinde kullanılan bir diğer kriptografik temel dijital imza algoritmalarıdır. Yaygın olarak ECDSA tercih edilir. Biz durumu genel bir dijital imzalama algoritması üzerinden verelim. Blok zinciri ağını kullanacak her bir kullanıcı bir önce kendi gizli anahtarını belirler ve bu gizli anahtar ile ilişkilendirilmiş açık anahtarı hesaplar. Burada gizli anahtarın belirlenmesi bir rasgele sayı

üretici yardımıyla olacaktır. Cüzdan uygulamaları farklı kriptografik araçları kullanarak bu işlemi kullanıcılar için doğrudan yapar. Buradaki en önemli detay gizli anahtarın saklanmasıdır. Bu durum kullanıcının sorumluluğunda olsa dahi bazı kriptografik çözümlerde üretilmiştir [52]. Daha sonra her bir kullanıcı uygun bir özetleme fonksiyonu ya da fonksiyonları kullanarak açık adresinin özet (ya da özetin özetleri şeklinde) değerini hesaplayarak kullanıcı adresini oluşturur. Böylece bu kullanıcıya işlem transfer etmek isteyen kullanıcılar bu adresi kullanacaklardır. Özetlersek bir A kullanıcısı:

$$Pr_A \rightarrow Pu_A \rightarrow H(Pu_A)$$

adımları takip edilerek adres oluşturur. Burada not edilmelidir ki bu adresler farklı blok zinciri teknolojilerinde farklı kodlama yapıları kullanılarak tutulurlar. Örneğin Bitcoin de bir açık anahtar önce SHA256 ile özetlenirken bu özet sonra RIPEMD160 ile özetlenir, yani

$$\text{Adres:} = \text{RIPEMD160}(\text{SHA256}(K))$$

şeklinde elde edilir. Fakat adresler bu şekilde görülmez. Örneğin Bitcoin’de adresler Base58Check kodlama algoritması ile tutulur. Ayrıca açık anahtarlar da önemlidir dolayısıyla diğer kullanıcılar tarafından bilinmelidir. Benzer şekilde açık anahtarlarda kodlanır ve çeşitli ön eklemeler yapılarak bu anahtarlara anlamlar katılır. Mesela Bitcoin özelinde açık anahtar bir eliptik eğri noktasıdır ama depolamak adına sadece x-bileşeni tutulur ve y bileşeni hakkındaki bilgi kodlamaya ön ek eklenerek belirtilir. Böylelikle saklanması gereken anahtarın boyutu düşürülmüş olur. Detaylar için [52] kaynağı incelenebilir.

Blok zincirinde asimetrik kriptografideki anahtar çiftlerinin nasıl kullanıldığını görmüş olduk. Fakat tek kullanım bu değildir. Bu anahtarlar aynı zamanda dijital imzalama algoritmalarında kullanılmak üzere de üretilmişlerdir. Dijital imzalama algoritmaları ise gerçekleştirilen işlemlerde kullanılmaktadır. Şimdi bu yapıdan detaylıca bahsedelim.

Daha önceki bölümlerde genel olarak bahsettiğimiz işlemler biraz daha detaylı ele alındığında, bir blok zinciri işlemi, bir kullanıcının diğerine bir dijital varlığı göndermesi sonucu oluşturulan veri olarak düşünülebilir. Dolayısıyla önce gönderilecek bir veri sonrada gönderilmiş olan veri olması gerekir. Bu durumlara işlemin girdileri ve çıktıları adı verilir.

- **Girdiler:** Göndericinin adresi, açık anahtarı, önceki işlemlerden harcanabilir bakiyesi (Unspent Transaction Output- UTXO) ve bu verilerin gönderici özel anahtarı ile oluşturulmuş dijital imzası.
- **Çıktılar:** Alıcının adresi, gönderilen dijital varlığın miktarı ve UTXO dakalan bakiye varsa, gönderenin kendisine iade edilen dijital varlığın miktarı.

Görüleceği üzere dijital imzalar bir işlemin olmazsa olmazıdır. Unutulmamalıdır ki bazı blok zinciri teknolojilerinde farklı yapıda işlemler de vardır. İşte blok zinciri yapılarında çift harcamayı engelleyen en önemli yenilik UTXO olarak gösterilebilir. Aslında bir işlem başlar ve son noktasına kadar ağaçlanarak bir zincir yapısı oluşturur. Burada dikkat edilmesi gereken durum bir işlemde UTXO daki tüm miktar gönderilirken istenilen miktar alıcıya aktarılır artan miktar ise göndericinin adresine geri gönderilir. Dolayısıyla her işlemde UTXO sıfırlanır sonra yeniden belirlenir.

Bir işlemin döngüsü Bölüm 4’te daha detaylıca anlatılmıştır. Burada önemli olan bir işlemin geçerliliği kontrol edilirken imza doğrulaması yapılması ve UTXO daki bakiyenin yeterliliğinin kontrol edilmesidir. Yani bakiye yetersiz ama imza doğru olsa işlem geçersizdir, ya da bakiye yeterli ama imza geçersiz olsa da işlem geçersizdir.

Bir işlem, bir veya birden fazla dijital varlığı girdi olarak alırken çıktı olarak ise yine bir veya birden fazla varlık oluşturulabilir. Burada denge önemlidir toplam girdi ve toplam çıktı varlık değerleri eşit olmalıdır. Bu durumda bir tek istisna vardır, gönderici ileride anlatacağımız madencinin veya hissedarın alabileceği bir işlem ücreti de belirleyebilir. Böyle bir işlemde toplam çıktı, girdiden az olacaktır aradaki fark ise işlem ücreti olacaktır. Bu durum tamamen opsiyonel gönderici tercihinin kalmıştır. Aşağıdaki şemalarda bazı tipteki işlem yapıları gösterilmiştir. Genel bir işlem:



Bu durumda fazla bakiye Çıktı2 olarak göndericinin adresine geri gönderilmektedir.



Bu yapı genelde bir kullanıcının birden fazla kullanıcıya (biri geri kendi adresi de olabilir) varlık gönderdiği yapıdır.



Bu yapı ise bir kullanıcının varlıklarını bütünlemesi olarak düşünülebilir. Çok yaygın bir tür olmamakla beraber işlemleri uygulamalar veya cüzdanlar yardımıyla yapan kullanıcıların yapmış olduğu işlemlerdir.

Ayrıca son iki yapının kombinasyonu şeklinde bir işlem gerçekleştirmek mümkündür.

Kriptografik protokoller kullanılarak iki farklı adresten tek adrese gönderim yapma durumları da gerçekleştirilmiştir. Bu tür işlemler çoklu imzalamalar gibi araçlar kullanılarak yapılır [53].

3.5.3. Fikir Birliği (Konsensüs) Algoritmaları

Şimdiye kadar blok zincirinin yapısı ve genel çalışma prensiplerinden bahsettik. Özetleme fonksiyonlarının ve dijital imzaların bu yapıların temel taşları olduğunu gördük. Fakat blok zinciri yapısının nasıl oluşacağına dair detaylardan bahsetmedik. Bu bölümde bir şekilde genesis blok oluşturulduktan sonra diğer bloklar nasıl oluşturulup özetleme fonksiyonu yardımıyla zincire eklenecek, sorusunun cevabını vereceğiz. Biliyoruz ki blok zincirleri merkezi olmayan yani bir üçüncü taraf tarafından yönetilmeyen yapılardır. Dolayısıyla böyle bir yapı yoksa blok ekleme işlemi nasıl gerçekleşmelidir, sorusu önem arz etmektedir.

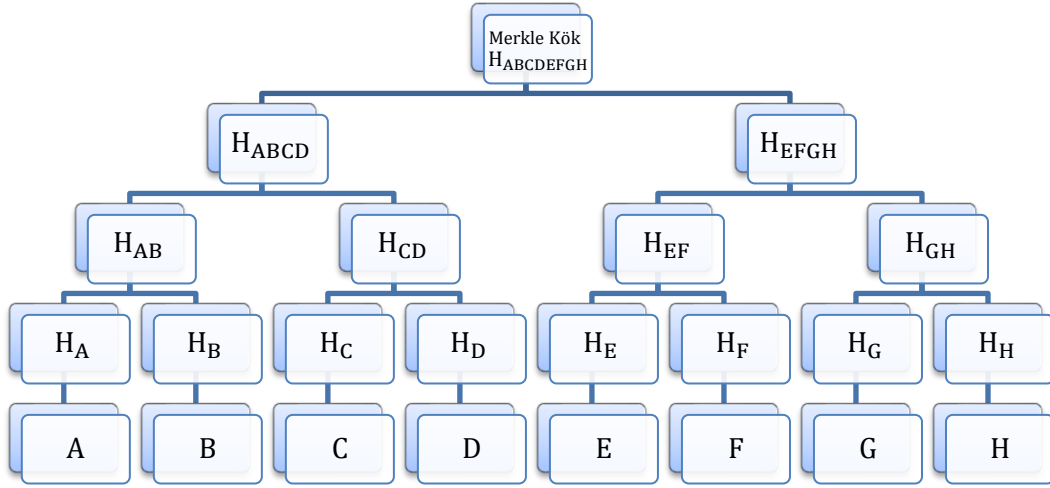
Bu soruya fikir birliği algoritmaları ile çözüm üretilmiştir. Fikir birliği algoritmaları ağdaki uç birimlerin doğruluk ve güvenilirlik konusunda anlaşmasını sağlayan mekanizmalar olarak tanımlanabilir. Bu mekanizma sayesinde sistem otoriteye ihtiyaç duymadan kullanıcılar tarafından çalışır bir halde kullanılır ve devamlılık sağlar. Fakat her kullanıcı sistemin iyi çalışması niyetinde olmayabilir. Yani sistemi bozmak isteyen, kötüye kullanmak isteyen kullanıcılar da olacaktır. Bu mekanizma uzun süredir bilgisayar biliminde dağıtık yapılara ilham olmuş bir problemi temel alarak kötü niyetli kullanıcıların sisteme zarar verme durumunun belli şartlarda ihmal edilebilir bir durum olduğuna dayanmaktadır.

Bizans Hata Toleransı Problemi: Problem tarihi bir hikâyeye sahiptir. Hikâye şuradan gelmektedir: Bir grup Bizans generali, bir şehre saldırmak için koordinasyon sağlamaya çalışıyor. Fakat generaller farklı konumlarda ve doğrudan iletişim kuramıyorlar. İletişim bir haberci (iyi niyetli) aracılığıyla kurulabiliyor. Ancak bazı generaller kötü niyetli olabilir ve yanlış bilgi yayabilirler. Eğer koordineli bir saldırı olmaz ise yenilecekler. İşte problem buradan doğarak ortaya çıkıyor. Güvenilir generallerin kötü niyetli generallerin varlığına rağmen ortak bir karar alabilmesi mümkün müdür? [54] Sorunun cevabı evet ama her zaman değil. Güvenilir generallerin sayısının toplam general sayısının $2/3$ 'ü kadar olduğu durumlarda ortak bir karar alabilmek mümkün olacaktır.

Dolayısıyla bu problemdeki fikirden yola çıkarak bir blok zincirinde, yeteri sayıda iyi niyetli güce sahip kullanıcı sayısı olduğu durumda otoriteye gerek kalmadan kötü niyetli kullanıcılara rağmen ortak karar alınabileceği değerlendirilmiştir.

Bu yapılardan detaylıca bahsetmeden önce bir veri yapısı türü olan Merkle Ağaçlarından bahsedilecektir [55]. Biliyoruz ki blok zinciri içerisinde ki bloklarda işlemler ilgili bloğun verileri olarak tutulmaktadır. Bu bloğun zincire eklenmesi verilerin doğru ve bir bütün olarak kontrol edilebilir olması lazımdır. İşte burada bize Merkle Ağacı yapıları yardımcı olur. Genel olarak Merkle ağacı, büyük miktarda verinin her birinin parçasının özet değerleri ve bu özet değerlerinin özetler zincirini tutarak bu verilerin doğruluğunu ve bütünlüğünü sağlamak için kullanılır. Aşağıdaki görsel ile bu yapıyı özetleyebiliriz.

Şekil 22: Merkle ağaç yapısı



Burada en altta A, B, C, D, E, F, G ve H verileri özet değerleri ile bir üst adıma yazılırken iki tane alt yaprağı olan üst yaprak bu iki alt yaprağın yan yana yazılarak özet değeridir yani

$$H_{AB} = H(H_A || H_B)$$

olarak hesaplanır. Burada en üst satıra ilgili Merkle ağacının kökü adı verilir ve kolayca görülür ki bu değer tüm verinin bilgisinin özeti olarak düşünülebilir. Dolayısıyla çok sayıda veri için bu şekilde bir özet değeri hesaplamak mümkündür. Bloklar içindeki işlemler için her bir blok için o bloğun Merkle kök değeri hesaplanır ve bloktaki veriler o değer ile temsil edilebilir. Merkle ağaçlarının bilgi ispatları ve kuantum sonrası kriptografide özetleme fonksiyonu tabanlı imzalama algoritmaları gibi daha birçok uygulaması vardır [56].

3.5.3.1. İş İspatı (PoW)

İş ispatı aslında bir bilgi ispatı protokolüdür. Yapılan bir hesaplamanın doğru yapıldığına dair sunulan ispattır. Bu ispat değerine Nonce değeri adı verilir. Yapılan işlem istenilen özellikte bir özet değeri hesaplamak için girdiye eklenmesi gereken değeri bulmaktır.

Bir blok zincirinde bir blok için bir önceki bloğun özeti ve bloktaki veriler, kısaca onların Merkle kökü, bloktaki en önemli verilerdir. Dolayısıyla iş ispatında bunlarında rolü olmalıdır. Teknik olarak aşağıdaki gibi tanımlayabiliriz. H bir özetleme fonksiyonu olmak üzere

PBH: Bir önceki bloğun özet değeri

MRH: Bloktaki verilerin Merkle root değeri

için amaç verilen bir zorluk derecesi n için

$$H(Nonce || PBH || MRH) = \underbrace{0 \dots 0}_{n \text{ tane}} * \dots *$$

şartını sağlayacak Nonce değerini bulmaktır. Yani özet çıktısının ilk n biti 0 olacak şekilde girdideki Nonce değerini belirlemek. Denk olarak özet değerinin tam sayı karşılığının 2^L değerinden küçük olması şeklinde de yorumlanabilir. Burada L değer özetleme fonksiyonun verdiği sabit bit uzunluğundan n değerinin çıkarılması ile elde edilir. Özetleme fonksiyonlarının doğası gereği girdi hakkında yorum yapmak zordur. Dolayısıyla Nonce değerini doğrudan belirlemek mümkün değildir. Kaba kuvvet araması yapılarak deneme yöntemiyle belirlenmektedir. Dolayısıyla çok fazla sayıda hesaplama gerekmektedir. Nonce değerini bulan uç birimlere bildirir ve onaylandıktan sonra blok ekleme işi gerçekleşir ve bir sonraki blok için bu işlem devam eder. Bu hesaplamaları yapıp Nonce değerini arayanlara madenciler adı verilir. Tabiki bu değeri arayan bir madenci değildir aynı anda birden çok madenci blok ekleyebilmek için uygun bir Nonce değeri bulma yarışı içerisinde. Yüksek hesaplama demek yüksek enerji tüketimi demek olduğundan madencilerin bir maliyeti vardır. Bu Nonce değerini ilk belirleyen madenci bir ödül alır ve ilgili digital varlığın üretimi bu şekilde gerçekleşmiş olur. Yani blok eklenmesi sonucu o varlıktan yenileri üretilmiş olur ve madencinin adresine iletilir. Dolayısıyla madencilerin bir motivasyonu mevcuttur. Ama yarışı bir kişi kazanacağından o blok için yarışan diğer madencilerin yaptığı tüm işler boşa gitmiş olacaktır. Bu durum PoW mekanizmasının darboğazı olarak görülmektedir.

Bu konuyla ilgili bazı temel nitelikteki soru ve cevaplara aşağıda değinilmektedir

- Aynı anda ya da benzer zaman diliminde iki farklı blok önerilirse ne olur? Bu durumda çatallanma dediğimiz bir olay gerçekleşir ve iki blok aynı anda son bloğa paralel eklenir. Çoğunluk devam ettiği blok esas blok olarak devam eder. Ama bazı durumlarda topluluk karar veremez ve çatallanma sonucu iki farklı zincire bölünme olur bu durumda çoğunluğun takip ettiği esas blok zinciri olarak devam ederken diğeri de bir yeni zincir olarak devam eder. Bitcoin zincirinde Bitcoin-Cash bunun en önemli örneklerindendir.
- Merkeziyetsizlik nasıl sağlanır? Herkes ortak bir yarışın içindedir. Dolayısıyla kimse bunu yapmada üstün değildir. Tabi ki hesaplama gücü yüksek olanın şansı daha fazla

olacaktır. Ama o kişinin kötü niyetini gerçekleştirmesi için hesaplama gücünün %51'ini ele geçirmiş olması gerekir ki pratikte bu durum pek olası değildir. Tabi çok az kullanıcı zincirler için bu durumun gerçekleşmesi olası olabilir. Dolayısıyla bir zincirin merkeziyetsizliğinin gücünü kullanıcı ve madenci sayısı belirleyecektir. Eğer bir madenci kötü niyetli olup ona göre Nonce değeri bulursa kimse onu takip etmez ve onaylamaz dolayısıyla o değeri bulana kadar yaptığı enerji tüketimi onun kaybı olacaktır. İşte bu maliyet madencileri iyi niyetli olmaya zorlamaktadır.

Yukarıda bahsettiğimiz çatallanmalar doğal olarak ortaya çıkan çatallanmalar olarak adlandırılır. Bir de istenilerek gerçekleştirilen çatallanmalar vardır. Bunlar zincirde önemli değişiklikler yapılması durumunda ortaya çıkar. Kullanıcılar bazen eskiyi tercih edebileceğinden topluluk tarafından yapay bir çatallanma yapılarak isteyenler eski zincire bağlı olanı tercih ederken, diğerleri yeni güncellemelerle oluşan zinciri takip edebilir. Bunun en önemli örneği 15 Eylül 2022 de Ethereum'ın PoW mekanizmasından bir sonraki alt bölümde anlatacağımız PoS mekanizmasına geçişte gerçekleştirdiği çatallanmadır.

Bu konunun avantajlarının yanında dezavantajları da mevcuttur. PoW mekanizmasının avantajlarını aşağıdaki gibi listeleyebiliriz.

- Merkeziyetsizlik amacına mükemmele yakın hizmet eder.
- Teknolojik gelişmeler sonucu işlem gücü yüksek cihazlar çıksa bile güvenlik seviyesi artırılarak blok ekleme süresi dengelenebilir.
- Sybill atak olarak bilinen sahte kullanıcıların sistemde hükmü olmamasını sağlar çünkü o kişiler gücünü de ikiye katlayamayacaktır.
- 2009 yılından beri Bitcoin ile başlayan pratik uygulamada işe yaradığı ispat edilmiştir.

Bunların yanında önemli dezavantajları da vardır:

- Doğası gereği yavaştır. Dolayısıyla bir işlemin gerçekleşmesi yani bloğa eklenmesi biraz zaman alacaktır.
- Doğal çatallanmalara yol açabilir.

- Sınırsız üretim olmaması için ödüller belli zamanlarda azaltılır. Bu durumda madencilerin motivasyonunu düşürür. Dolayısıyla kullanıcılar madencileri motive etmek adına daha yüksek işlem ücretlerine katlanmak durumunda kalabilir.
- En önemlisi ise boşa giden o kadar hesaplama aslında boşa giden o kadar enerji tüketimi demektir. Yani fazlasıyla gereksiz enerji tüketimine neden olmaktadır. Bu da bu mekanizmayı kullanmanın doğaya zarar verdiği görüşüne sebep olmuştur.

Bu gereksiz enerji tüketimi araştırmacıları yeni fikir birliği mekanizmalarına yönlendirmiştir. Bunlardan ikincil en yaygını ve çalıştığı ispat edilmiş olanı Hisse ispatı (Proof of Stake- PoS) mekanizmasıdır.

3.5.3.2. Hisse İspatı (PoS)

Çevre dostu merkeziyetsizliği hedefleyen bir fikir birliği algoritması olarak ortaya atılmıştır [57]. PoW mekanizmasında enerji tüketiminin fazla olması bu mekanizmanın gelişmesine sebep olmuştur. Blok oluşturma işi bir yarış olarak değil de sistem tarafından belirlenen bir kişi tarafından yapılmaktadır. Bu işi yapanlara hissedarlar (staker) adı verilir.

Sistemin çalışmada öncelikle kullanıcının topluluk tarafından belirlenen bir dijital varlık miktarından fazla varlığı olmalıdır. Bu varlık sistem tarafından kitlenir (stake) ve o kişi hissedar olur. Genel yapıda bu miktar sabittir. Ama PoS farklı varyasyonlarında değişik yapılar mevcuttur. Daha sonra sistem eşit şansla bu kişilerden birini rasgele seçer ve bu kişi blok eklemek için Merkle kökü bulup verileri sisteme ekler. Doğrulayıcılar bu bloğu doğruladıktan sonra diğer blok ekleme işlemleri benzer şekilde devam eder. Eğer hissedar kötü niyetliyse doğrulayıcılar doğrulamaz ve kitlenmiş olan dijital varlığını kaybeder. Buda iyi niyetli olmanın maliyeti olarak görülebilir. Hissedarların motivasyonu da yine ödüdür. Fakat ödül sistemi blok zinciri teknolojisine göre farklılık göstermektedir. Ethereum PoS için ödül sistemine [58] kaynağından ulaşılabilir.

Avantajları aşağıdaki gibi listelenir:

- Enerji tüketimi çok çok azdır. Dolayısıyla daha sürdürülebilirdir.

- İşlem hızı çok daha yüksektir. Hesaplamaya dayalı bir zaman kaybı yoktur.
- Dijital varlığını kaybetme riski insanları iyi niyetli olmaya zorlar.

Dezavantajları ise:

- PoW'a göre daha az pratikte gözlemlenmiştir.
- Sybill atağa karşı dayanıklı değildir. Varlığı çok olan farklı kullanıcı hesapları ile daha çok hissedar rolü alabilir. Yani bir kişi bir den fazla hissedar rolünde olabilir. Bu da varlığı çok olanın sistemi kötüye kullanabilme riskini oluşturur. Görece büyük zincirler için uygundur ama yeni başlamış bir zincirde riskler barındırır.

Burda detaylarından bahsetmeyeceğimiz Practical Byzantine Fault Tolerance (PBFT), Delegated Proof Of Stake (DPoS), Proof of Capacity, Proof of History vs. gibi birçok fikir birliği algoritması mevcuttur.

3.5.4. Blok Zinciri Teknolojisinin Bazı Sıkıntıları

Her teknolojik gelişme gibi blok zincirlerinin de bazı sıkıntılı yanları vardır. Biz burada teknik ve pratik anlamda bu sıkıntıların bazılarını ele alacağız.

Öncelikler en büyük sıkıntı blok zincirinin boyutunun tüm verilerin kaydedildiği bu zincir uç birimler tarafından saklandığından boyut katlanarak artmaktadır. Bu da uç birimler için yüksek depolama alanları gerektirmektedir. Bunun için hafif uç birimler gibi daha az veri tutan kullanıcı çözümleri türetilmiştir. Ayrıca Mina blok zinciri ZK-Snark protokolünü rekürsif bir şekilde kullanarak 22kB bir zincirdir. Bitcoin zinciri 650 GB boyutlara yaklaşmışken, Ethereum zinciri 1.25 TB boyutu aşmış durumdadır. Ayrıca katman çözümleri de bu boyut problemine çözüm olarak türetilmişlerdir. Bu durum aynı zamanda Bölüm 2.3.11 de bahsedilen ölçenebilme problemini de ortaya çıkarmaktadır.

Bir diğer husus ise hız konusudur PoS gibi mekanizmalar hızlı çözümler üretirken halen istenilen seviyelerde değildir. PoH kullanan Solana günümüzdeki en hızlı blok zinciri teknolojilerinden biridir. Fakat bu yapılar merkeziyetsizlikten ödün vermektedir. Dolayısıyla merkeziyetsizliği PoW kadar ön planda tutup daha hızlı işlem gerçekleştirmeyi sağlayacak yapıların geliştirilmesi veya araştırılması önem arz etmektedir.

Son olarak en önemli eksik blok zincirinin dijital varlık dışında kullanılabilecek olup olmadığının günümüzde halen tam olarak belirlenememiştir. Dezavantajlarından dolayı tercih edilip edilmemesi sorusuna cevap vermek oldukça zordur. Bunun için yerel çözümler adına özel blok zinciri yapıları geliştirilmiştir. Bu tür zincirler otorite tarafından yönetilmektedir. Bu durum ise blok zincirinin ortaya çıkış felsefesiyle uyuşmamaktadır.



4. BLOK ZİNCİRİ TEKNOLOJİSİNİN ÇALIŞMA MODELİ

Bu bölümde, Bölüm 3’de teknik detaylarından bahsettiğimiz yapıların Bitcoin ve kısmen Ethereum özelinde daha detaylı olarak incelenmesi yapılacaktır.

4.1. Modelin Bitcoin Özelinde Ayrıntılı İncelenmesi

Bu bölümde blok zinciri çalışma sürecinin Bitcoin özelinde nasıl işlediği ayrıntılı olarak anlatılacaktır. Bitcoin örneğinin merak eden okuyucular için de oldukça faydalı bir örnek olacağı kanaatindeyiz. Bitcoin C++ dilinde yazılmıştır ve açık kaynak kodludur. Ayrıca bu yapılar gerekli yerlerde Bitcoin kaynak kodları referans olarak gösterilerek açıklanacaktır. Burada Bitcoin Core’un v25.0 sürümü kullanılmaktadır ve bu sürüm Mayıs 2024 tarihinde yayımlanmıştır.

Bitcoin sisteminde yapılacak güncellemeler için BIP’ler (Bitcoin Improvement Proposal) oluşturulmaktadır. BIP, Bitcoin’e yeni özellik eklemek, protokolü değiştirmek ya da teknik standartlar belirlemek için önerilen belgeler bütünüdür. Oluşturulan BIP’ler topluluk ve geliştiriciler tarafından tartışılır. Bitcoin Core’un güncel versiyonu oluşturulurken kabul gören bu BIP’ler de uygulamaya alınır. Aşağıda bazı önemli BIP’ler ile onları uygulamaya alan Bitcoin Core versiyonları yer almaktadır.

Tablo 5: Bazı Bitcoin Core sürümleri ve uyguladığı BIP’ler

Bitcoin Core Sürümü	Uyguladığı BIP’ler	Açıklama
0.10.0 (2015)	BIP 66	DERS imza standardı zorunlu hâle getirildi
0.12.1 (2016)	BIP 68, 112, 113	CHECKSEQUENCEVERIFY ve zaman kilitleri
0.13.0 (2016)	BIP 152	Compact Blocks (hızlı blok yayılımı)
0.13.1 (2016)	BIP 141, 143, 147	SegWit protokolü tanıtıldı

0.16.0 (2018)	BIP 173	Bech32 adres formatı tanıtıldı
0.20.0 (2020)	BIP 157/158	Neutrino protokolü (light clients)
0.21.0 (2021)	BIP 324 (hazırlık)	Şifreli P2P bağlantılar için temel
22.0 (2021)	BIP 340, 341, 342	Taproot, Schnorr ve MAST aktivasyonu
26.0+ (2023-2025)	BIP 324, 329	Şifreli bağlantılar + Cüzdan etiketleme

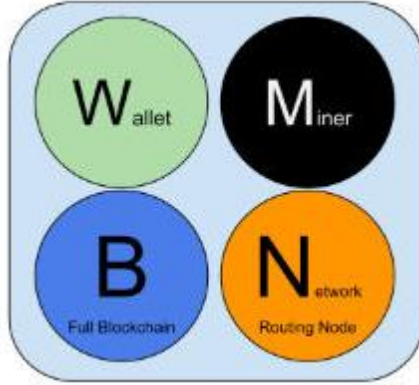
Bitcoin merkeziyetsiz bir sistem olduğu için çalışmasını sağlayan ve süreci takip eden merkezi bir yapı ya da sunucu yoktur. Yazılımı GitHub ya da bitcoin.org adresinden indirilebilmektedir. Bitcoin.org Bitcoin Topluluğu tarafından yönetilen anonim bir yapıdır. GitHub ise açık kaynak kodlu projelerin hem geliştirilmesini hem de dağıtımını sağlayan bir platformdur. Bitcoin Core yazılımını indirip bilgisayarına kuran herkes artık sistemin hem çalıştırıcısı hem işleticisi hem de kullanıcısı olmaktadır. Bu kapsamda merkeziyetsizliğin temeli olan düğüm(node) konusunu ayrıntılı olarak incelememiz gerekecektir.

Kaynaklarda düğüm kavramı tam düğüm (full node) ve hafif düğüm (light node) olarak adlandırılrsa da konuyu incelediğimizde Bitcoin kaynak kodunda ve mimarisinde hafif düğüm gibi bir kavram yer almamaktadır. Satoshi makalesinde basitleştirilmiş ödeme doğrulamasından (Simplified Payment Verification– SPV) bahsetmektedir. Bu düğüm olmayan kullanıcıların Bitcoin ağında doğrulama yapabilmelerini sağlayan bir mekanizmadır. Günümüzde hafif düğümler ise üçüncü taraflar tarafından geliştirilmiş SPV yöntemini kullanan yazılımlardır. Bu açıdan bakıldığında ne ağın güvenliğine ne mekiyetsizliğe katkıları bulunmamaktadır. Bu açıdan hafif düğüm isimlendirmesinin içeriğe tam uymadığı kanaatindeyiz. Hafif düğümler donanımı düşük ve bağlantısı zayıf kullanıcıların ekosistemde bulunması açısından önemlidir. İlerleyen kısımda düğüm kavramını Bitcoin Core yazılımını indiren tam düğümler için kullanılacaktır.

Düğümler kendi başlarına çalışan bağımsız yapılardır. Bazı kaynaklar düğümleri dört farklı yapı altında incelemektedir. [11] Bu yaklaşım Bitcoin topolojisini ortaya koymak adına daha anlaşılır bir görüntü sunmaktadır. Ancak bir düğümün kısımları çok daha fazla başlık altında

incelenebilmektedir. Yaygın kullanılan bu dört başlık “Cüzdan”, “Madencilik”, “Tüm Blok Zinciri” ve “İletişim Katmanı” olarak adlandırılabilir. Burada birçok farklı yapı “İletişim Katmanı” altında toplulaştırılmıştır. İlerleyen kısımda bütün bu başlıklara ve yapılara ayrıntılı olarak yer verilecektir.

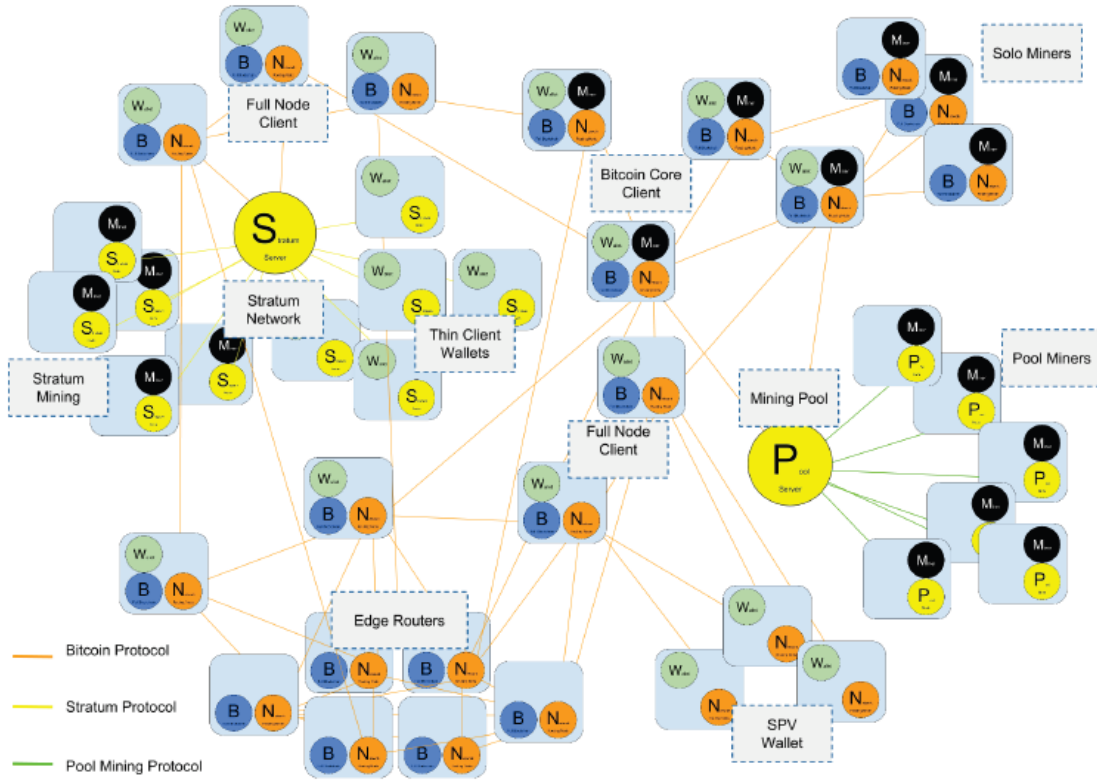
Şekil 23 : Tam bir bitcoin düğümü.



Kaynak: Mastering Bitcoin sayfa 140

Bu gösterim aşağıda yer alan bitcoin ağ yapısının anlaşılmasında oldukça kullanışlı olmaktadır. Görüldüğü üzere bazı düğümlerde madencilik kısmı bazı düğümlerde ise cüzdan kısmı bazı düğümlerde de her ikisi de yer almamaktadır. Bitcoin core yazılımını yükleyerek bir düğüm olduğunda cüzdan ve madencilik kısımları aktif etmek ya da deaktif etmek kullanıcının tercihi bırakılmaktadır. Blok zincirinin tamamı ve iletişim katmanı ise tam düğüm olarak adlandırılan düğümlerde aktif olmak zorundadır. Aşağıdaki şekilde görülen diğer düğüm tipleri ise Bitcoin orijinal kodunda ve mimarisinde yer almayan ancak zaman içerisinde ortaya çıkan ihtiyaçlar sonucu ortaya çıkmış basit düğümlerdir. Bu düğümlerin merkezizetsizlik ya da dağıtıklık anlamında sisteme bir katkıları yoktur. Ancak uygulama süreçlerini kolaylaştırmak için oluşturulmuşlardır. Burada kırmızı ile gösterilen ağ orijinal Bitcoin ağı yeşil ve sarı ile gösterilen ağlar ise daha sonradan uygulama kolaylığı oluşturmak için basit düğümler arasında kullanılan ağlardır. Bu yapılar sonunda tam bir düğüme bağlanarak sisteme erişim sağlarlar. Burada yer alan SPV ağda tam bir düğüm çalıştırmadan doğrulama yapılabilmesi için kurgulanmış bir yöntemdir ve Bitcoin whitepaper’ının 8. Bölümünde anlatılmaktadır.

Şekil 24 : Genişletilmiş bitcoin ağı.



Kaynak: Mastering Bitcoin Sayfa 143

Yeni bir tam düğüm sisteme katıldığında ilk iş olarak ağda yer alan diğer düğümlerle iletişim kurarak onlardan bilgi almaya başlar. Daha sonra iletişim kurulan bu düğümlerden bütün blok zinciri veritabanı ve UTXO veritabanı indirilir. Diğer düğümlerden işlem havuzu (Mempool) bilgileri alınır. Tüm bloklar indirildikten ve işlem havuzu güncellendikten sonra düğüm tam kapasite ile çalışmaya başlayabilmektedir.

Düğümler Bitcoin sisteminin yapıtaşlarını oluştururlar sayıları ne kadar fazla olursa sistem o kadar güvenli ve dağıtık olmaktadır. Sistemin amacı mümkün olduğunca fazla kullanıcının düğüm olmayı tercih etmesidir. Ancak bu durum hem kullanılacak ekipmanlar hem de harcanacak enerji anlamında kullanıcıya maliyet oluşturmaktadır. Bu sorunu çözmek için sistem düğüm olmayı avantajlı hale getirmeye çalışmıştır. Düğüm olan kullanıcı bütün verileri doğrulayabilmektedir. Bütün karar alma süreçlerine dahil olmaktadır. Opsiyonel olarak gelen cüzdan uygulamasının aktif hale getirilmesi sonrasında düğüm olan kullanıcı kendi cüzdanını ve hesabını yaratarak başka bir uygulamaya ihtiyaç duymadan işlem

gerçekleştirebilmektedir. Yine düğüm olan kullanıcı opsiyonel olarak gelen madencilik modülünü aktif hale getirerek madencilik yapabilmektedir. Madencilik Bitcoin yaratma yöntemidir. Yaratılan Bitcoin'ler madencilere ödül olarak dağıtılır. Yine yapılan işlemler için ödenen komisyonlar da madencilere dağıtılarak madencilik dolayısı ile kullanıcıların düğüm olmayı tercih etmesi teşvik edilmektedir.

Bitnodes sitesine göre, bu dataya bakıldığı gün itibarıyla dünya üzerinde 7 tanesi de Türkiye'de bulunan toplam 19523 [11] aktif düğüm yer almaktadır.

Düğümün içindeki yapılar, birbirleri ile sürekli iletişim halinde olmakla birlikte yukarıda bahsettiğimiz başlıklar kapsamında ayrı ayrı anlatılacak daha sonrasında bu birimlerin birbirleri ile iletişimleri incelenecektir.

Bir düğüm tam blok zincirini barındırır. Bu Bitcoin'in ilk oluşturulduğu günden bugüne kadar olan süreçteki bütün blokları kapsar. Bitcoin'de bir blok "Blok Başlığı" ve "İşlem Listesinden" oluşur. Blok başlığı Önceki Blok Hash'i, Merkle Kökü, Zaman Damgası, Hedef Zorluk, Nonce değeri ve Versiyon bilgisinden oluşur.

Kod 1: Bitcoin blok başlığı parametreleri

```
cpp Kodu kopyala  
  
1  class CBlockHeader  
2  {  
3  public:  
4      // Blok başlığı bileşenleri  
5      int32_t nVersion;           // Blok versiyonu  
6      uint256 hashPrevBlock;     // Önceki bloğun hash değeri  
7      uint256 hashMerkleRoot;   // Merkle ağacının kök hash değeri  
8      uint32_t nTime;           // Zaman damgası (Unix Epoch formatında)  
9      uint32_t nBits;           // Hedef zorluk (compact format)  
10     uint32_t nNonce;          // Madencilikte kullanılan nonce değeri
```

Kod:src/primitives/block.h

Blokların zincir oluřturması iin gereken baėlantı blok bařlıklarının hash'lenmesi (bařlıkta yer alan bilgilerin SHA256 algoritması ile iki kere hash'lenmesi) sonucu elde edilen hash deėerinin bir sonraki bloėun bařlıėına yazılmasıyla saėlanmaktadır. Bylece retilen her yeni blok ilk bloktan bařlayarak birbirine deėiřtirilemeyecek bir algoritma ile baėlanmış olur. Bu durum Bitcoin'de iřlemlerin deėiřtirilemezliėini saėlamaktadır.

Bařlıkta yer alan versiyon deėeri blok formatı ya da protokolde yapılan deėiřiklikleri, gncellemeleri ifade etmektedir. Nonce (number used Once) istenen formatta hash deėeri hesaplayabilmek iin deėiřtirilen tek alandır. Madenciler bu deėer iin rastgele deėerler deneyerek matematiksel problemi zmekte ve yeni blok oluřturabilmektedirler.

Merkle Kk bir veri setinin doėruluėunu kontrol etmek iin kullanılan bir algoritmadır. İřlem havuzuna gelen iřlemlerden seilerek oluřturulacak yeni blok da yer alan iřlem listesi bu algoritma kullanılarak bir hash deėerine indirgenir ve bu deėer blok bařlıėına yazılır. Bu sayede blokta yer alan btn iřlemlerin hi deėiřtirilmemiř olduėu garanti edilir.

Kod 2: Merkle Kk deėerini hesaplayan kod parası

```
16 uint256 CBlock::BuildMerkleTree() const {
17     std::vector<uint256> leaves;
18     for (const auto& tx : vtx) {
19         leaves.push_back(tx->GetHash());
20     }
21     return ComputeMerkleRoot(leaves, nullptr);
22 }
```

Kod:src/primitives/block.cpp

Burada iřlem listesinin Merkle Kk deėerini hesaplayan kod parası grlmektedir. Merkle algoritmasının nasıl alıřtıėını gsteren kod parası ise ařaėıda yer almaktadır.

Kod 3: Merkle algoritmasının nasıl çalıştığını gösteren kod parçası

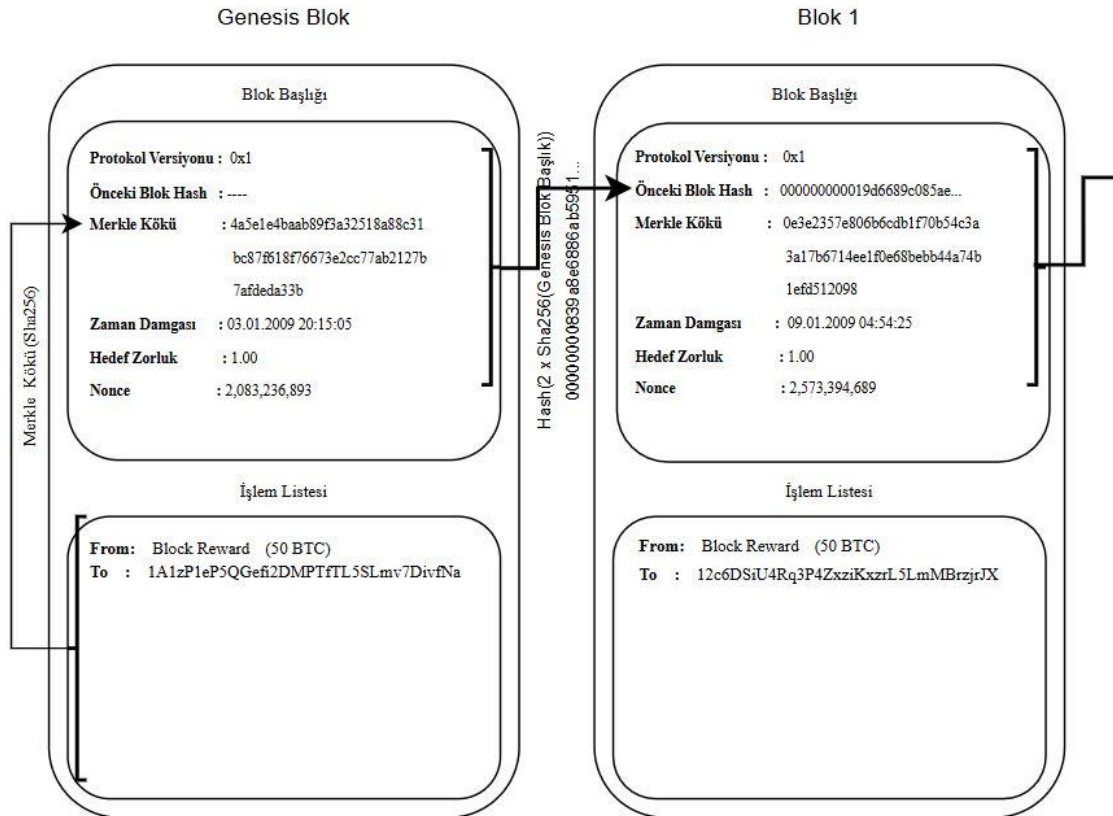
```

9  uint256 ComputeMerkleRoot(const std::vector<uint256>& leaves, bool* mutated) {
10     if (leaves.empty())
11         return uint256();
12
13     bool has_mutation = false;
14     std::vector<uint256> hashes(leaves);
15     while (hashes.size() > 1) {
16         if (mutated && hashes.size() % 2 == 1) {
17             has_mutation = true;
18             hashes.push_back(hashes.back());
19         }
20         std::vector<uint256> new_hashes;
21         for (size_t i = 0; i < hashes.size(); i += 2) {
22             new_hashes.push_back(Hash(hashes[i], hashes[i+1]));
23         }
24         hashes = new_hashes;
25     }
26     if (mutated) *mutated = has_mutation;
27     return hashes[0];
28 }

```

Kod: src/consensus/merkle.cpp

Şekil 25: Bitcoin de blok yapısı



Yukarıdaki şekilde konunun daha iyi anlaşılabilmesi için Başlangıç (Genesis) bloğu ve Blok1 gerçek değerleri ile verilmiştir. Ocak 2009 tarihinde ilk blok oluşturulmuş. Henüz ilk bloklar oluşturulduğu için sadece sistem tarafından 50 BTC verilmiş ve bu işlem Merkle ağacının ilk yaprak düğümünü oluşturmuştur. Her blokta işlem listesi sistem tarafından ödül verilme işlemi (coinbase) ile başlar. Daha sonra işlemlerin hash değerleri bulunur. Bu hash değerleri 2 li olarak toplanır ve yeni hash değerleri bulunur. Sonunda tek hash hesaplanana kadar bu işlem serisi devam ettirilmektedir. Elde ettiğimiz son hash değeri merkle kökü olarak başlığa yazılır. Burada her hash alma SHA256 algoritmasının iki defa uygulanması ile gerçekleştirilir. Böylelikle blok başlığının bütün değerleri tamamlanmış olur ve madenciler bu başlığın zorluk derecesine uygun şekilde hash değerini bulmaya çalışırlar. Bulunan bu değer bir sonra oluşturulacak olan bloğun başlığına yazılır. Böylelikle değiştirilemez bir zincir yapısı oluşturulmuş olmaktadır.

Bir blok sığası maksimum 1 MB olacak şekilde kurgulanmıştır. Ancak zaman içinde işlem hacminin artması ile bu sığa yetersiz duruma gelmiştir. Yapılan tartışmalar ardından Bitcoin ağında Segregated Witness (SegWit), 2017 yılında Bitcoin Improvement Proposal (BIP) kapsamında tanıtılmış ve 23 Ağustos 2017 tarihinde etkinleştirilmiştir. Yeni formatta oluşturulmuş bloklar segwit blok olarak adlandırılırken klasik yöntemle oluşturulmuş bloklar legacy blok olarak adlandırılmaktadır. Bu konu Bitcoin de yapılan güncellemeleri anlamak adına oldukça güzel bir örnek olarak değerlendirilebilir. Blokların sığalarının yetersizliği sorunu gündeme gelmiş Bitcoin platformlarında tartışılmış ve çoğunluğun kararıyla bir soft-fork (kısmi çatallanma) tercih edilmiştir. Bu tercih sonucu olarak yeni yaklaşımı kullanan düğümler ile eski yaklaşımı kullanan düğümlerin işleyişinde herhangi bir sorun olmamaktadır. Burada yaklaşım işlemlerin gönderi alıcı ve tutar bilgilerini orijinal blok içine yazarken anahtarlar ve imza bilgileri ayrı bir blokta tutulur. Bu yaklaşım bir blok içine yazılabilecek işlem sayısını artırmaktadır. Segwit ile işlemlerin boyutları değil ağırlığının hesaplanması kavramı yaratılmış ve Bitcoin kaynak kodunda güncellemeler yapılmıştır.

Kaynak kodda segwit öncesi 1MB olan sığa güncelleme sonrası 4MB ağırlığa ulaşabilir hale getirilmiştir. Segwit ile verinin büyüklüğü değil ağırlığı hesaplanmaya başlanmıştır.

Kod 4: Segwit güncellemesi ile ilgili tanımlanan yeni değişkenler

cpp

Kopyala Düzenle

```
21 static const unsigned int MAX_BLOCK_SERIALIZED_SIZE = 1000000; // 1 MB sınırı
```

cpp

Kopyala Düzenle

```
22 static const unsigned int MAX_BLOCK_WEIGHT = 4000000; // 4 MB sınırı
23 static const unsigned int WITNESS_SCALE_FACTOR = 4; // Tanık verilerinin ölçek faktörü
```

src/consensus/consensus.h dosyasında yapılan değişiklikler

Kod 5: Segwit güncellemesi ile blok ağırlık hesaplama yöntemi

cpp

Kopyala Düzenle

```
1920 int64_t GetBlockWeight(const CBlock& block)
1921 {
1922     return ::GetSerializeSize(block, SER_NETWORK, PROTOCOL_VERSION) +
1923         WITNESS_SCALE_FACTOR * ::GetWitnessSize(block);
1924 }
```

Kod: src/primitives/transaction.cpp

Böylelikle bir blok 4MB ağırlığa kadar ulaşabilmektedir.

Yaklaşık 10 dk da bir bir blok üretilmektedir. Bu sürenin neden 10 dk seçildiği net olarak bilinmemekle birlikte daha hızlı blok üretiminin üretilen blokların ağda yayılması konusunda sorunlar oluşturabileceği ve hızlı blok üretimi ile çatallanma olasılıklarının artacağı değerlendirilmektedir. Yeni bir blok oluşturma sürecinde, madencilerin yaptıkları hesaplamalarda nonce değerini değiştirerek istenen hash değeri bulmaları tamamen olasılıksal bir süreçtir. Sistem her blok için 10 dk da oluşturulma garantisi verememektedir. Bu durum hesaplanacak işlemin zorluğunun ayarlanması ile çözülmektedir. Böylelikle bir süre çok hızlı istenen hash değeri hesaplayan madenciler bu değeri bulma sürecinin zorlaştırılmasıyla daha yavaş istenen değeri bulabilmekte ve uygulanan matematiksel

yaklaşım sayesinde bu süre ortalama olarak 10 dk ya yaklaşmaktadır. Zorluk ayarlama süreci ise 2016 blokta bir yani yaklaşık 2 hafta da bir gerçekleştirilmektedir. Zorluk hesaplaması şu şekildedir:

Hedef Süre=2016×600=1,209,600 saniye (14 gün).

Gerçek Süre=Son Blok Zamanı–İlk Blok Zamanı.

Yeni Zorluk=Mevcut Zorluk × Gerçek Süre / Hedef Süre

Kod 6: Blok için süre değişkenleri

```
1 18 static const int64_t nPowTargetSpacing = 10 * 60; // 1 blok için hedef süre (10 dakika)
2 19 static const int64_t nPowTargetTimespan = 14 * 24 * 60 * 60; // 2016 blok için hedef süre (14 gün)
3
4 21 // Zorluk Ayarlama: Hedef zaman dilimi (14 gün) her 2016 blokta bir uygulanır.
5 22 static const int DifficultyAdjustmentInterval() { return nPowTargetTimespan / nPowTargetSpacing; }
```

Kod: src/consensus/params.h

Kod 7: Blok için zorluk hesaplaması

```
1 unsigned int GetNextWorkRequired(const CBlockIndex* pindexLast, const CBlockHeader *pblock,
2                                 const Consensus::Params& params)
3 {
4     assert(pindexLast != nullptr);
5     unsigned int nProofOfWorkLimit = UintToArith256(params.powLimit).GetCompact();
6
7     // Sadece zorluk ayarlama aralığında değişiklik yap
8     if ((pindexLast->nHeight+1) % params.DifficultyAdjustmentInterval() != 0)
9     {
10         if (params.fPowAllowMinDifficultyBlocks)
11         {
12             // Testnet için özel zorluk kuralı
13             if (pblock->GetBlockTime() > pindexLast->GetBlockTime() + params.nPowTargetSpacing*2)
14                 return nProofOfWorkLimit;
15             else
16             {
17                 // Son özel olmayan minimum zorluk bloğunu döndür
18                 const CBlockIndex* pindex = pindexLast;
19                 while (pindex->pprev && pindex->nHeight % params.DifficultyAdjustmentInterval()
20                     != 0 && pindex->nBits == nProofOfWorkLimit)
21                     pindex = pindex->pprev;
22                 return pindex->nBits;
23             }
24         }
25         return pindexLast->nBits;
26     }
27
28     // 14 günlük bloklara geri git
29     int nHeightFirst = pindexLast->nHeight - (params.DifficultyAdjustmentInterval()-1);
30     assert(nHeightFirst >= 0);
31     const CBlockIndex* pindexFirst = pindexLast->GetAncestor(nHeightFirst);
32     assert(pindexFirst);
33
34     return CalculateNextWorkRequired(pindexLast, pindexFirst->GetBlockTime(), params);
35 }
```

Kod:

src/pow.cpp

Yukarıda kodlarda da gösterildiği üzere zorluk asla eski halinin 4 katından fazla artırılamaz, kolaylaştırma işlemi de 4 katından daha fazla kolaylaştırılmaz.

Geçerli Hedef Hash = Maksimum Hedef Hash / Zorluk seviyesi

$$\text{Maksimum Hedef Hash} = 2^{256-1} \approx 1.158 \times 10^{77}$$

Hesaplanan zorluk seviyesi ile geçerli hedef hash hesaplanmış olur. Bu değerin binary gösteriminde oluşturulan 256 bitlik sayının başındaki sıfır sayısı bulunacak bloğun zorluğunu belirlemektedir.

Böylelikle bir bloğun temel özelliklerini ve birbirine bağlanarak nasıl blok zinciri yapısı oluşturduğunu incelemiş olduk. Bir düğüm bütün bu blok zinciri verisini içinde barındırmaktadır.

Bitcoin cüzdanları düğümler tarafından ya da üçüncü taraf şirketler tarafından yazılan uygulamalarla oluşturulmaktadır. Bitcoin core yazılımı çalıştırılarak oluşturulmuş bir düğüm de cüzdan yazılımı inaktif olarak gelmektedir. Kullanıcı bu yazılımı aktif hale getirerek bitcoin transferi gerçekleştirebilmektedir. Bitcoin Core kullanıcılara GUI (Graphical User Interface) ve CLI (Command Line Interface) olmak üzere iki farklı yöntemle düğüm ile iletişim sağlama imkanı sunmaktadır. CLI ile metin tabanlı ama daha fazla özelleştirme ve kontrol imkanına sahip olunmaktadır. GUI ise grafik tabanlı olması sayesinde kullanıcı dostudur ve komut yazmayı gerektirmez ancak CLI kadar esnek ve güçlü değildir. Bu iletişim metodlarından biri kullanılarak Bitcoin cüzdanı aktif hale getirilebilmektedir.

Bir cüzdanın nasıl çalıştığını anlayabilmemiz için birçok farklı bilginin anlaşılması gerekmektedir. Bunların başında UTXO (Unspent Transaction Output) gelmektedir. Bitcoin sisteminde hesabınıza gelen Bitcoin'ler toplanarak bir bakiye oluşturulmamaktadır. Her işlem çıktısı biriciktir ve bu haliyle bir işleme konu edilebilir. Böylelikle yaratılan bir Bitcoin'in tüm yaşam döngüsü takip edilebilir. Bu konuyu anlamak için en uygun

metaforun çek olduđu kanaatindeyiz. Bitcoin sürekli yeniden yazılan ve harcanan çekler gibi çalışmaktadır. Ancak bu çekler sisteme ait bir yerde tutulmaktadır ve burada harcanan çekler yok edilirken yeni yazılan çekler yine burada depolanmaktadır. A ve B kişilerinin C kişisine sırasıyla 1 ve 2 Bitcoin gönderdiğini düşünelim. Bitcoin sistemi C'nin hesabında 3 bitcoin olduđu kaydını düşmez. C'ye ait 1 tane 1 Bitcoin'lik ve 1 tane 2 Bitcoin'lik çeki var gibi kayıt gerçekleştirir. C'nin D'ye 0,5 Bitcoin gönderdiğini düşünelim. C sisteme bu işlemi hangi çekle yapacağını söylemek zorundadır. Ya da bu işi cüzdanlar C için otomatik olarak yapar. Örneğimiz de bu işlemin 2 Bitcoin'lik çek ile yapıldığını düşünelim. Bu işlem onaylandığında sistem C'ye ait 2 Bitcoin'lik çeki yok edecektir. D adına düzenlenmiş 0,5 Bitcoin'lik bir çek ve yine C'ye geri kalan hesap için C adına düzenlenmiş 1,5 Bitcoin'lik bir çek düzenleyecektir. Sistem sürekli bu mantıkla çalışmaktadır. Bazı işlemler için 10'larca bazen 100'lerce çek kullanmak gerekebilmektedir. Bu yüzden Bitcoin sisteminde bir cüzdan olmadan işlem yapmak oldukça zor ve teknik bilgi gerektiren bir süreçtir. Örneğimizde geçen çek kavramını UTXO olarak düşünebiliriz. Sistem ilk bloktan beri oluşturulan UTXO'ların kaydedildiği ve harcanan UTXO'ların silindiği bir liste oluşturmaktadır. Her işlem sırasında harcanmak istenen UTXO'ların bu listede olup olmadığı sorgulanır. Blok doğrulandığında ise harcanan UTXO'lar bu listeden çıkarılırken yeni eklenen UTXO'lar listeye eklenmektedir. Böylelikle aynı UTXO'nun 2 kere harcanmasının önüne geçildiği gibi her UTXO'nun kime ait olduđu bilindiğinden başkası tarafından harcanması da engellenmektedir.

Kod 8: Bitcoin’de Coin değişkeninin yapısı

```
cpp Kopyala Düzenle  
  
class Coin {  
public:  
    CTxOut out; // Çıkış (miktar ve scriptPubKey içerir)  
    unsigned int nHeight; // UTXO’nun ait olduğu blok yüksekliği  
    bool fCoinBase; // Coinbase işlemi olup olmadığını belirler  
  
    // Constructor (kurucu) tanımları  
    Coin(CTxOut&& outIn, int nHeightIn, bool fCoinBaseIn)  
        : out(std::move(outIn)), nHeight(nHeightIn), fCoinBase(fCoinBaseIn) {}  
  
    Coin() : nHeight(0), fCoinBase(false) {}  
  
    // Fonksiyonlar  
    bool IsCoinBase() const { return fCoinBase; }  
    bool IsSpent() const { return out.IsNull(); }  
};
```

Kod: src/coins.h

Kod 9: Bitcoin’de UTXO yapısı

```
cpp Kopyala Düzenle  
  
class CTxOut {  
public:  
    CAmount nValue; // Çıkışın değeri  
    CScript scriptPubKey; // Çıkışı harcamak için gereken script  
  
    CTxOut() { SetNull(); }  
    CTxOut(const CAmount& nValueIn, CScript scriptPubKeyIn)  
        : nValue(nValueIn), scriptPubKey(std::move(scriptPubKeyIn)) {}  
  
    void SetNull() { nValue = -1; scriptPubKey.clear(); }  
    bool IsNull() const { return nValue == -1; }  
};
```

src/primitives/transaction.h

Kod parçaları incelendiğinde UTXO’nun Bitcoin sisteminde coin olarak tanımlandığı görülmektedir. Bir UTXO 4 farklı bilgi taşımaktadır. Bunlar; Çıkış miktarı(amount), ait

olunan kişinin genel anahtarı (Script), blok yüksekliği yani kaçınca blokda oluşturulduğu (height), madencilikle üretilip üretilmediğidir (CoinBase).

Bitcoin adreslerinin, genel ve özel adreslerin nasıl oluşturulduğu konusu da burada değinmemiz gereken önemli bir konudur. Özel anahtar (Private Key) 256 bitlik rastgele bir sayıdır. Yani 256 tane 0 veya 1 in yan yana gelmesiyle oluşur. 8 bit 1 byte olduğu için aslında bu 32 byte lık bir sayıdır. Bu sayının gizli tutulması gerekir çünkü bütün işlemler bu anahtar ile gerçekleştirilmektedir. Bu özel anahtardan eliptik Eğri Kriptografisi (ECC) kullanılarak secp256k1 adı verilen eğri ile genel anahtar üretilir. Burada özel anahtardan her zaman ilgili kriptografik yöntem kullanılarak genel anahtar üretilebilirken genel anahtardan özel anahtar pratik değildir. Genel anahtar sıkıştırılmış (33byte uzunluğunda 0x02 ya da 0x03 ile başlayan) ya da sıkıştırılmamış (65byte uzunluğunda 0x04 ile başlayan) formatta olabilir. Genel Anahtar SHA256 algoritması ile hashlenerek 256 bitlik bir hash değeri elde edilir. Daha sonra RIPEMD-160 algoritması ile 160 bitlik bir değer elde edilir. Bu değere tamamlanmamış adres diyelim. Çünkü adresin tamamlanması için bir ön ek ve sonuna bir kontrol eki eklenecektir. Bu aşamada bu hesabın türünü belirten bir ön ek eklenir. Versiyon öneki tablosu aşağıda yer almaktadır.

Tablo 6: Bitcoin adresleri versiyonları.

Adres Türü	Versiyon Bayrağı	Başlangıç Karakteri	Açıklama
Legacy (P2PKH)	0x00	1	Genel anahtar hash'ine ödeme.
Script Hash (P2SH)	0x05	3	Script hash'ine ödeme.
Bech32 (SegWit)	bc1	bc1	SegWit adresleri.
Testnet Legacy (P2PKH)	0x6F	m veya n	Test ağı için genel anahtar hash'i.
Testnet Script Hash (P2SH)	0xC4	2	Test ağı için script hash'i.
Testnet Bech32 (SegWit)	tb1	tb1	Test ağı için SegWit adresleri.

Versiyon eklenmiş tamamlanmamış adres değeri 2 kere SHA256 algoritması ile haslenerek elde edilen hash değerinin ilk 4 byte değeri kontrol eki olarak ön ekli tamamlanmamış adres değerinin sonuna eklenir. Daha sonra Base58Check kodlamasına ile bitcoin adresi elde edilmiş olur.

Adres oluřturma srecinin olduka fazla iřlem gerektirdiėi ve birok adrese ihtiya duyulabileceėi dřndėnde adres oluřturma iřini elle yapmanın hata yapma riskini ok artıracakėı ařıktır. Bu yzden czdan kullanmak bir tercihten ziyade zorunluluk haline gelmektedir.

Dėm zerinde aktif hale getirilmiř bir czdan; zel anahtar, genel anahtar, adresleri, UTXO listesi, iřlem gemiři, yedekleme bilgileri gibi birok bilgiyi tutar. Bu czdanlar iřlem doėrulamalarını bařka hibir uygulamaya ihtiya duymadan gerekleřtirebilirler. Diėer czdanlar ise doėrulama yapmak iin bir tam dėme baėlanmak zorundadırlar. Bir czdan asla Bitcoin tutmaz aslında btn Bitcoin'ler yalnızca UTXO listesinde bulunurlar. Bunun dıřında bir yere tařınamazlar kopyalanamazlar. Czdanlar zel anahtarları saklarlar ve bu anahtarla harcanabilecek UTXO'lar ile iřlem yapmanızı saėlarlar.

Bir czdan ncelikle zel ve genel anahtar doėrulaması yapar. Adres geerliliėini doėrular, iřlemde kullanılan UTXO'ların varlıėı ve geerliliėini doėrular. İřlem imzalarını doėrular. Eėer bir iřlem czdan zerinden gnderilmediyse bu kontroller iřlemin gnderildiėi dėm tarafından gerekleřtirilir. Burada czdanların teknik olarak zorunluluk olmadėı anlařılmaktadır. Ancak czdan dıřı yapılacak iřlemler hem ok karmařık hem de olduka yksek hata yapma riski ierdiėinden czdan kullanmadan iřlem yapmak kullanıřsız grlmektedir.

Bir dėmn bir bařka nemli fonksiyonu ise madenciliktir. Madencilik 3 temel fonksiyonun gerekleřtirilmesini saėlamaktadır. Bunlardan birincisi iřlemlerin doėrulanarak blok zincirine eklemesidir. İkinci nemli fonksiyon ise Bitcoin retimidir. Son olarak ise bu sistem Bitcoin aėının gvenliėini saėlamaktadır.

Madenciler yukarıda ayrıntılı anlattėımız zere oluřturulan bir blok sonrası bu bloktan retilen hash deėeri bilgisi ile hemen yeni bir hash deėeri retmek iin alıřmaya bařlar. Burada nceki bloėun hash deėeri, zaman damgası, bu bloėa yazılacak iřlemlerin merkle kk deėeri, zorluk seviyesi ve nonce deėerinin SHA256 algoritması ile hash deėeri

hesaplanır. İstenilen zorluk seviyesine uygun hash değeri bulunana kadar (örneğin hesaplanacak hash değerinin ilk 19 basamağı sıfır isteniyor olabilir.) nonce değeri değiştirilerek yeniden hash hesaplanır. Bu süre 2016 blokta bir hesaplanan zorluk değeri ile 10 dk civarında tutulmaktadır.

Bitcoin mimarisinde madenciliğin düğümler tarafından yapılması kurgulanmış olsa da zaman içerisinde madencilik havuzlarının oluşması ile düğümlerin solo madenciliği işlevsiz hale gelmiştir. Bitcoin genişletilmiş ağ yapısına bakıldığında bu madencilik havuzları da birer hafif düğüm olarak gösterilmektedir. Günümüzde madencilik, bir düğüm olup matematiksel hesaplama yapmak yerine bir madencilik havuzuna kendi donanımınızın hash hesaplama gücünü kiralamanız şeklinde gerçekleşmektedir.

Madenciler işlemleri mempool denilen bir havuzdan alırlar. Bir işlem bir düğüme geldiğinde gerekli doğrulamalara tabi tutularak bu havuza gönderilir. Madenciler de bu havuzdan çektikleri işlemleri doğrular ve diğer parametreleri de ekleyerek bir blok oluştururlar ve bloğun hash değerini hesaplarlar. Burada bazı kontroller tekrar edilmektedir. Yapılan işlemin düzgün formatta oluşturulması ve imzaların doğrulanması kontrolü hem mempool hem de madenci tarafından yapılmaktadır. UTXO kontrolü mempool tarafından girdilerin harcanmamış UTXO listesinde yer alıp almadığına bakılarak yapılır. Madenci hem UTXO listesine bakar hem de mempool da yer alan işlemlerin çıktıları ile olan eşleşmeleri inceler. Çift harcama kontrolü hem mempool hem de madenci tarafından kontrol edilir ve aynı UTXO kullanan işlemler oluşursa yüksek ücretli olan seçilir. Bağımlılık ve blok boyutu kontrolü madenciler tarafından yapılmaktadır. Bağımlılık bir işlemin çıktısının başka bir işlemin girdisi olarak aynı blokta yer alması durumudur. Bu durumda sıralama önem arz eder ve madenci bu sıralamaya dikkat eder. Ücret kontrolleri düğümün kendi kural setine göre değişiklik gösterebilir. Bazı düğümler minimum ücret tutarını karşılamayan işlemleri reddetmektedir. Madenciler ise en yüksek ücretli işlemleri öncelerler.

Madencilik Bitcoin mimarisinde tamamen otomatik olarak yapılan bir süreçtir. Ancak Bitcoin'in zaman içerisinde değerinin artması sonucu madencilğe olan ilginin artması bu alanda insanların daha farklı cihazlarda istenen hash değerini bulma çabasını tetiklemiştir. Bu durum farklı cihazların sisteme adapte edilmesi için yeni yazılımların yapılması ve

sistemin ayarlarının güncellemesi, sistemin donanımsal (soğutma, enerji vs.) ve yazılımsal (yamalar, güncellemeler vs.) ihtiyaçlarının karşılanması gibi daha maliyetli ve karmaşık süreçlerin doğmasına sebep olmuştur. Sonrasında çok büyük işlem gücüne sahip madencilik havuzlarının ortaya çıkması bireysel olarak yapılan madenciliği kullanışsız hale getirmiştir. Havuzların işlem gücü karşısında bir kullanıcının işlem gücü değerlendirildiğinde ekstra donanım kullanmadan madencilik yapması matematiksel olarak anlamsız hale gelmektedir.

Madencilik düğümlerde opsiyonel olarak gelen bir özelliktir. Her düğüm madencilik yapmak zorunda değildir. Madencilik Bitcoin Core da komut satırı üzerinden (CLI) veya RPC API'sini kullanarak aktif hale getirilebilir. Madencilik modunun açılmasından sonra sistem otomatik olarak madencilik yapmaya başlar. Blok oluşturulduktan sonra hemen yeni blok için bir şablon oluşturulur. Mempool dan işlemler seçilir ve kontroller yapılır. Nonce değeri sürekli değiştirilerek blok başlığının SHA256 algoritması iki kere kullanılarak hedef zorluğa uygun bir hash değeri hesaplanmaya çalışılır. Bulunan değer sonrası blok bağlantılı düğümler üzerinden ağa yayımlanır. Diğer düğümler tarafından doğrulanan blok, blok zincire eklenir ve madencilik sistemi hemen yeni bir blok için şablon oluşturulur. Bu döngü sürekli devam eder.

Zorluk seviyesine uygun bir özet değerinin hesaplanması ve bloğun onaylanması ile bu değeri bulan madencinin hesabına sistem tarafından coinbase denilen özel bir işlemle blok ödülü ve işlem ücretleri gönderilir. Bu işlem bir blokta yer alan ilk işlemdir. Madenciliği teşvik amaçlı olarak verilen bu ödül aynı zamanda Bitcoin yaratma biçimidir. Bu süreç bu yüzden madencilik olarak anılmaktadır.

Bitcoin core kaynak koduna baktığımızda COIN sabitinin 100.000.000 birim olarak tanımlandığı görülmektedir. Bir Bitcoin aslında 100.000.000 satoshiden oluşan bir birimdir. Yine koda baktığımızda blok ödülünün 50 COIN yani 50 Bitcoin olduğunu görmekteyiz. Ancak halving değişkeni hesaplanarak bu ödülün belirli zaman aralıkları ile yarıya indirildiği görülmektedir. Bu süre 210.000 blok olarak belirlenmiştir. Yani her 210.000 blokta bir sistem ödül olarak verdiği tutarı yarıya indirmektedir. Bu yarılanma olayının 64 sefer ile sınırlandırıldığı görülmektedir. Ancak 33. halving de ödülün 1 satoshi'nin altına düşecek olması sebebiyle toplam arzın 21.000.000 olacağı hesaplanmaktadır.

Kod 10: Bir Bitcoin 100.000.000 Satoshi'dir.

```
cpp Kopyala Düzenle  
  
/** Amount in satoshis (Can be negative) */  
typedef int64_t CAmount;  
  
static const CAmount COIN = 100000000; // 1 Bitcoin = 100,000,000 satoshi
```

Kod: Src/amount.h

Kod 11: Halving süreci ile ilgili kod parçaları

```
cpp Kopyala Düzenle  
  
CAmount GetBlockSubsidy(int nHeight, const Consensus::Params& consensusParams) {  
    int halvings = nHeight / consensusParams.nSubsidyHalvingInterval;  
  
    // Maksimum halving sayısı  
    if (halvings >= 64) {  
        return 0;  
    }  
  
    // Blok ödülü hesaplama  
    CAmount nSubsidy = 50 * COIN;  
    nSubsidy >>= halvings; // Ödülü yarıya indir  
    return nSubsidy;  
}
```

Kod: validation.cpp

```
cpp Kopyala Düzenle  
  
consensus.nSubsidyHalvingInterval = 210000; // Her 210,000 blokta bir halving
```

Kod: Src/amount.h

Tablo 7: Bitcoin halving tablosu

Halving Sayısı	Blok Yüksekliği	Tarih	Blok Ödülü BTC	Toplam Bitcoin BTC
0 (Başlangıç)	0	3 Ocak 2009	50	0
1. Halving	210,000	28 Kasım 2012	25	10,500,000
2. Halving	420,000	9 Temmuz 2016	12.5	15,750,000
3. Halving	630,000	11 Mayıs 2020	6.25	18,375,000
4. Halving	840,000	20 Nisan 2024	3.125	19,687,500

Burada belirtmeliyiz ki halving sonrası blok ödülleri azalıyor olması madencilik yapmayı karsız hale getirebilir. Çünkü madencilik artık ciddi yatırım yapılan ve kayda değer enerji harcayan bir yapı haline gelmiştir. Bu konuda sistemin beklentisi Bitcoin'in yaygınlaşması sonucu kullanımının ve değerinin artması ile işlem ücretlerinin madencileri tatmin edici seviyeye gelmesidir.

Madenciler büyük miktarda hesaplama gücü kullanarak zorluk hedefini karşılayan bir hash bulmaktadır. Bu durum sistem güvenliğini sağlamaktadır çünkü geçerli olmayan işlemleri sisteme kabul ettirmeye ya da sistemi bir şekilde manüpile etmeye çalışacak kötü niyetli aktörlerin çok büyük miktarda işlem gücüne sahip olmalarını ve büyük miktarda enerji tüketmelerini gerekli kılmaktadır. Çifte harcamalar ise madencilerin işlemleri tekrar kontrol etmeleri ile en uzun ve en fazla iş kanıtına sahip zincirleri kabul etmeleriyle önlenmektedir. Madenciler mempool dan çıktıkları işlemleri tekrar doğrulayarak işlem güvenliğini sağlarlar.

Düğümün 4 başlık altında incelendiği mimaride dördüncü kısım bütün iletişimin sağlandığı ve doğrulamaların yapıldığı iletişim katmanıdır. Aslında bu kısım birçok alt başlıkta incelenebilir. Öncelikle bir düğümün diğer düğümlerle iletişimi incelenecektir.

Bir düğüm bitcoin.conf dosyası kullanılarak kullanıcı tarafından yapılandırılabilir. Düğüm bağlantı ayarları, RPC erişimini etkinleştirme, cüzdan işlevselliğini açıp kapama, madenciliği etkinleştirme, veri depolama ve performans ayarları optimize edilebilir.

Yeni kurulan bir düğüm DNS Seed hizmeti üzerinden ağına bağlı diğer düğümlerin IP adresini sağlar ve onlarla bağlantı kurar. Bu bağlantılardan blokları alır ve ilk bloktan (Genesis Blok) başlayarak her bloğu doğrular. Bütün zincir indirilip doğrulandığında tam senkronize bir düğüm haline gelir. Bu süreçten sonra sistemin bir parçası olarak yaşamına devam eder. Bitcoin Core, varsayılan olarak 125 aktif bağlantıyı desteklemektedir. Bu sayı kullanıcı tarafından değiştirilebilir. Düğüm yeni düğümler (peer) keşfeder ve onlara bağlanır.

Düğüm peer'lerden gelen işlemleri alır ve doğrular. Geçerli işlemleri mempool'a ekler. Yine peer'lerinden gelen blokları alır ve doğrular. Geçerli blokları kendi zincirine ekler ve diğer düğümlere yayar.

Bir düğüm bir çatallanma (fork) durumuyla karşılaşrsa en uzun zinciri kabul eder ve alternatif zincirleri reddeder. Çatallanma, aynı anda birden fazla geçerli blokun üretilmesiyle geçici olarak birden çok zincir uzantısının ortaya çıkması durumudur. Bu genellikle ağdaki düğümlerin farklı blokları aynı anda alması sonucu oluşur. Ancak zamanla madencilerden biri zinciri daha fazla blokla uzattıncaya, daha uzun olan zincir "en fazla iş yapılan" zincir olarak kabul edilir. Diğer zincirler geçerliliğini yitirir ve yetim blok (orphan block) olarak adlandırılır.

Bir düğüm her onaylanan blok sonrası UTXO setini günceller. Gelen işlemler sonrası mempool'da geçici UTXO listesi oluşturur.

Bir işlemin yaşam döngüsüne bakmanın süreçlerin daha anlaşılır olmasını sağlayacağı kanaatindeyiz. Bu sistemin temel yapı taşı olan işlemler, kullanıcının cüzdanından başlatılır ve işlem nesnesi farklı aşamalarda çeşitli formlar alır ve her aşamada belirli kontrollerden geçirilir. Kullanıcı tarafından başlatılan transfer talebi, cüzdan uygulaması üzerinden işlem yapısına dönüştürülür. Bu yapı, CMutableTransaction sınıfı ile temsil edilmekte olup işlemin imzalanmadan önceki düzenlenebilir hâlidir. Cüzdan bu aşamada, kullanıcı bakiyesini temsil eden UTXO havuzundan uygun girdileri (vin) seçer ve hedef adrese yönlendirilmiş çıktıları

(vout) oluşturur. İmzalanan işlemde değişiklik yapılamayacağı için imza öncesi işlemler üzerinde değişiklik yapılabilir form da tutulmaktadır.

Kod 12: Düzenlenebilir işlem parametreleri

```
class CMutableTransaction {  
public:  
    int32_t nVersion;  
    std::vector<CTxIn> vin;  
    std::vector<CTxOut> vout;  
    uint32_t nLockTime;  
};
```

Kod: src/primitives/transaction.h

İşlemin oluşturulması sırasında cüzdanın gerçekleştirdiği kontroller, kullanıcı tarafından başlatılan bir işlemi mantıksal ve ekonomik açıdan geçerli kılmak amacıyla yürütülen çeşitli yazılım düzeyinde doğrulamalardır. Bu kontroller işlemin ağa gönderilmeden önce geçerli ve kabul edilebilir olmasını sağlamak için kritik öneme sahiptir:

- **Harcanabilirlik Kontrolü:** Cüzdan, kullanıcının bakiyesini oluşturan UTXO'ları tarar ve seçilen UTXO'nun başka bir işlemde harcanıp harcanmadığını kontrol eder. Bu işlem wallet.cpp dosyasındaki AvailableCoins() fonksiyonu üzerinden gerçekleştirilir.
- **Yeterli Bakiye ve Input Seçimi:** İşlem için gerekli toplam miktarın karşılanabilmesi için uygun UTXO'lar seçilir. Bu seçim, en az sayıda girdiyle işlem boyutunu minimize edecek şekilde kurgulanmıştır. Kodda SelectCoins() fonksiyonu bu işi üstlenir.
- **Fee Hesaplaması:** İşlemin boyutu tahmin edilerek güncel mempool koşullarına göre bir işlem ücreti belirlenir. Bu işlem EstimateSmartFee() ve GetMinimumFee() gibi fonksiyonlarla gerçekleştirilir. Eğer kullanıcı manuel olarak bir ücret belirlememişse, bu otomatik hesaplanır.
- **Change Output Oluşturulması:** Eğer input'lar toplamı, hedeflenen miktar ve işlem ücretinden fazla ise fazla bakiye kullanıcıya geri gönderilecek şekilde yeni bir çıktı (change output) olarak eklenir. Bu işlem CreateTransaction() fonksiyonu içinde yer alır.

- **İşlem Parametrelerinin Belirlenmesi:** İşlemin versiyonu (nVersion) ve geçerlilik zaman penceresini tanımlayan nLockTime değeri belirlenir. Bu parametreler, işlemin hangi bloktan itibaren geçerli olacağını belirler.

İşlem imzalanmadan önce, ilgili input için imzalanacak veri SignatureHash fonksiyonu ile hesaplanır. Burada işlemin kendisi değil hash değeri imzalanmaktadır. Bu işlem, input'un bağlı olduğu script türüne göre özel davranış sergiler (örneğin SegWit için farklı, Legacy için farklı hesaplama yöntemi vardır çünkü segwit de imza kısmı işlemin içinde değildir).

Kod 13: İşlemin imzalanacak olan hash'ini oluşturan fonksiyon

```
uint256 SignatureHash(const CScript& scriptCode, const CMutableTransaction& txTo,  
    unsigned int nIn, int nHashType, const CAmount& amount,  
    SigVersion sigversion);
```

Kod: src/script/interpreter.cpp

İmzalama işlemi SignSignature() fonksiyonu ile gerçekleştirilir:

Kod 14: İmzalama işlemi gerçekleştirilen fonksiyon

```
bool SignSignature(const CKeyStore& keystore, const CScript& scriptPubKey,  
    CMutableTransaction& txTo, unsigned int nIn, int nHashType);
```

Kod: src/script/sign.cpp

İmzalama işlemi tamamlandıktan sonra, işlem sabit (immutable) hale getirilerek CTransaction nesnesine dönüştürülür. Bu haliyle işlem, artık ağda yayılmaya ve zincire dahil edilmeye hazırdır.

Kod 15: Bir işlemin sabitlenmiş işlem parametreleri

```
class CTransaction {
public:
    const int32_t nVersion;
    const std::vector<CTxIn> vin;
    const std::vector<CTxOut> vout;
    const uint32_t nLockTime;
};
```

Kaod: src/primitives/transaction.h

İşlem, AcceptToMemoryPool fonksiyonu aracılığıyla mempool'a alınmadan önce kapsamlı doğrulamalardan geçirilir. Bu kontroller, işlemin ağ kurallarına ve yerel politika filtrelerine uygunluğunu denetler.

Kod 16: Bir işlemin havuza alınması ve yapılan kontroller

```
bool AcceptToMemoryPool(...) {
    if (!CheckTransaction(tx, state)) return false;
    if (!Consensus::CheckTxInputs(tx, state, view, GetSpendHeight(view))) return false;
    if (!IsFeeSufficient(tx)) return false;
    pool.addUnchecked(hash, entry);
    return true;
}
```

Kod: src/validation.cpp

Düğümün gerçekleştirdiği kontroller:

- İmzaların geçerliliği (EvalScript)
- UTXO'ların harcanabilir olup olmadığı (double-spend kontrolü)
- Gerekli minimum fee'nin karşılanıp karşılanmadığı
- Transaction formatının ve boyutunun uygunluğu

Madenciler, CreateNewBlock() fonksiyonu ile mempool'daki işlemleri değerlendirmeye alır. Hedefleri, blok sınırları içerisinde mümkün olan en yüksek fee toplamını elde edecek işlem setini oluşturmaktır.

Kod 17: Madencinin blok oluşturarak işlemleri kendi bloğuna seçmesi

```
std::unique_ptr<CBlockTemplate>
    BlockAssembler::CreateNewBlock(const CScript& scriptPubKeyIn) {
    while (HasMoreTransactions()) {
        AddToBlock(tx);
    }
    return pblocktemplate;
}
```

Kod: src/miner.cpp

Madencinin yaptığı kontroller:

- İmza doğrulaması (EvalScript)
- İşlemin daha önce dahil edilip edilmediği (duplicate kontrolü)
- Fee oranı değerlendirmesi (satoshi/byte)
- Blok boyutu sınırının aşılması

Yeni blok üretildiğinde, ConnectBlock() fonksiyonu ile zincire eklenir. Bu aşamada her işlem için harcanan UTXO'lar silinir ve yeni UTXO'lar sistem kayıtlarına eklenir.

Kod 18: İşlemlerin madenci bloğundan blok zinciri bloğuna yazılması

```
bool CChainState::ConnectBlock(...) {
    for (const auto& tx : block.vtx) {
        UpdateCoins(tx, view, nHeight);
    }
    return true;
}

void UpdateCoins(const CTransaction& tx, CCoinsViewCache& inputs, int nHeight) {
    // Harcanan UTXO'lar kaldırılır
    // Yeni çıktıları UTXO setine eklenir
}
```

Kod: src/validation.cpp

Bu süreç, Bitcoin'in merkeziyetsiz yapısının teknik olarak nasıl sürdürüldüğünü açık biçimde göstermektedir. Cüzdan uygulaması işlemi başlatırken yerel kontrolleri yürütür; ağdaki düğümler işlemi doğrular ve mempool'a alır; madenciler ise işlemleri değerlendirerek ekonomik optimizasyonla blokları oluşturur. Nihai olarak işlemler zincire

kaydedilir ve UTXO modeli güncellenir. Her bir katman, protokol bütünlüğünün korunmasında hayati bir rol oynar.

Bu yapının modülerliği, Bitcoin'in sansüre karşı dirençli ve doğrulanabilir bir sistem olarak çalışmasını mümkün kılmaktadır.

4.2. Ethereum ve Akıllı Kontratlar

Ethereum'un kurucusu Vitalik Buterin tarafından 2014 yılında yayımlanan “Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform” [59] başlıklı whitepaper incelendiğinde, Buterin'in Bitcoin teknolojisi konusundaki hakimiyeti ve Bitcoin'e yönelik getirdiği bazı eleştiriler dikkat çekmektedir.

Buterin'e göre, Bitcoin'in eşten eşe (peer-to-peer) çalışması ve merkez bankası ya da merkezi otoritesi bulunmayan bir para birimi olarak tasarlanması oldukça yenilikçi ve devrim niteliğinde bir gelişmedir. Ancak Bitcoin'in sağladığı yenilikler yalnızca bununla sınırlı değildir. Vitalik Buterin, Bitcoin'in ortaya koyduğu teknolojinin arkasındaki temel yapı olan blok zinciri konseptinin aslında ikinci ve belki de daha kritik bir yenilik olduğunu ifade etmektedir. İnsanların giderek bu teknolojiyi daha iyi anladığını ve kullanım alanlarının yalnızca para transferleriyle sınırlı olmadığını vurgulamaktadır.

Buterin'in Ethereum whitepaper'ında belirttiği üzere, blok zinciri teknolojisinin kullanım alanları, klasik finansal işlemlerin çok ötesine geçmektedir. Blok zinciri üzerinde dijital varlıkları temsil eden özel para birimleri ("colored coins"), merkezi olmayan finansal borsalar, finansal türev ürünleri, akıllı mülkiyet sistemleri ("smart property"), alan adları (Namecoin), merkezi olmayan kumar ve bahis platformları gibi pek çok yenilikçi uygulamanın mümkün olduğunu ortaya koymaktadır.

Blok zinciri teknolojisinin en önemli araştırma alanlarından biri olarak gösterilen akıllı sözleşmeler (smart contracts), belirli önceden tanımlanmış kurallar ve koşullar gerçekleştiğinde dijital varlıkları otomatik olarak hareket ettirebilen programlardır. Vitalik

Buterin, akıllı sözleşme kavramının bir adım ötesinin ise Merkezi Olmayan Otonom Organizasyonlar (Decentralized Autonomous Organizations- DAOs) olduğunu ifade etmiştir. Bu organizasyonlar, tüm varlıklarını ve kurallarını blok zinciri üzerinde şeffaf ve otomatik şekilde tutan, merkezî bir kontrol mekanizmasına ihtiyaç duymayan uzun vadeli akıllı sözleşmelerdir.

Ethereum'un whitepaper'da belirtilen temel hedefi ise, Bitcoin'in sınırlı betik dili (script language) yeteneklerini aşan, Turing-complete (evrensel hesaplama yeteneğine sahip) bir programlama dili içeren yeni nesil bir blok zinciri platformu oluşturmaktır. Ethereum sayesinde kullanıcılar, herhangi bir durum geçişini (state transition) tanımlayabilen ve diledikleri karmaşıklıkta uygulamalar oluşturabilecekleri akıllı sözleşmeleri yazabileceklerdir. Bu durum, yalnızca finansal işlemler için değil, finansal olmayan pek çok farklı uygulama türü için de sınırsız olanaklar sunmaktadır. Böylece Ethereum, yalnızca Bitcoin'in eksiklerini tamamlamakla kalmayıp blok zinciri teknolojisinin potansiyelini çok daha ileri seviyelere taşıma iddiasındadır.

Ethereum konusunu incelerken daha önce çok ayrıntılı olarak anlattığımız Bitcoin konusuyla karşılaştırmalı olarak ilerlemenin konunun anlaşılması için çok daha mantıklı olacağı kanaatindeyiz.

Öncelikle Bitcoin bir ödeme aracı olma iddiasıyla yola çıkmıştır [9]. Bitcoin'in Script dili, bilerek ve isteyerek, yalnızca basit "doğru/yanlış" (true/false) harcama koşullarını değerlendirmeye yönelik olarak sınırlandırılmıştır. Ethereum, genel amaçlı ve programlanabilir bir blok zinciri olarak tasarlanmıştır ve herhangi bir karmaşıklıkta kodu çalıştırabilen bir sanal makineye (Ethereum Virtual Machine- EVM) sahiptir. Bu sanal makine sayesinde Ethereum'un dili evrensel hesaplama yeteneğine sahip (Turing-complete) olduğundan, Ethereum herhangi bir genel amaçlı bilgisayar gibi doğrudan işlev görebilmektedir [60].

Ethereum'un para birimi Ether (ETH) dir. $1 \text{ ETH} = 10^9 \text{ Gwei} = 10^{18} \text{ Wei}$ 'dir (Bitcoin'de $1 \text{ BTC} = 10^8 \text{ Satoshi}$).

Ethereum 30 Temmuz 2015 tarihinde çalışmaya başladı ve Genesis blok oluşturuldu. Ethereum Genesis bloğunda 8.893 işlem bulunmaktadır. Bu işlemler Ethereum'un ilk halka arzı (ICO) sırasında fon sağlayan katılımcılara ETH dağıtımını içerir. Bu ICO, 22 Temmuz 2014'te başlamış ve 42 gün sürerek yaklaşık 31.591 Bitcoin toplamıştır. Bu, o dönemin kuru ile yaklaşık 18,4 milyon ABD dolarına ve toplamda 60.102.216 ETH'ye tekabül etmektedir [61].

Ethereum çok fazla çatallanma (hard fork) deneyimi yaşamıştır. Bu çatallanmaların birçoğu önceden planlı ve topluluk tarafından kabul gören çatallanmalardır. Ancak birinde toplulukta ayrışmalar olmuş bu sebeple yeni coin (ETC) oluşmuştur. Bu güncellenmeler liste halinde aşağıda yer almaktadır [62].

Tablo 8: Ethereum hard fork'ları

Hard Fork Adı	Tarih	Açıklama	Yeni Coin Oluşumu
Frontier	30 Temmuz 2015	Ethereum ana ağının ilk sürümü. Geliştiricilere ve madencilere ağ üzerinde deney yapma fırsatı sundu. Akıllı sözleşmeler oluşturulabiliyordu, ancak kullanıcı dostu arayüzler henüz mevcut değildi.	Hayır
Ice Age	1 Eylül 2015	Zorluk bombasını devreye sokarak madenciligi zamanla zorlaştırmayı ve böylece Proof of Stake (PoS) sistemine geçişi teşvik etmeyi amaçladı.	Hayır
Homestead	1 Mart 2016	Ethereum'un ilk istikrarlı sürümü. Ağ güvenliği ve işlem hızında iyileştirmeler yapıldı. Akıllı sözleşmelerin daha güvenli ve verimli çalışması sağlandı.	Hayır
DAO Fork	1 Temmuz 2016	DAO adlı merkeziyetsiz otonom organizasyonun saldırıya uğraması sonucu alınan fonları geri almak amacıyla gerçekleştirildi. Bu fork, Ethereum (ETH) ve Ethereum Classic (ETC) olmak üzere iki ayrı zincirin oluşmasına neden oldu.	Evet, Ethereum Classic (ETC)
Tangerine Whistle	1 Ekim 2016	Ağıdaki hizmet reddi (DoS) saldırılarına karşı savunma mekanizmalarını güçlendirdi ve gaz hesaplamalarında iyileştirmeler yaptı.	Hayır
Spurious Dragon	1 Kasım 2016	Daha fazla DoS saldırı vektörüne karşı önlemler aldı, ağın verimliliğini artırdı ve zararlı hesapları temizledi. Ayrıca, tekrar saldırılarına (replay attacks) karşı koruma mekanizmaları ekledi.	Hayır
Byzantium	1 Ekim 2017	Metropolis güncellemesinin ilk aşaması. Gizlilik ve güvenlik özelliklerinde iyileştirmeler yaptı, zk-SNARKs gibi teknolojilere destek sağladı ve akıllı sözleşmelerin yeteneklerini genişletti.	Hayır

Constantinople & St. Petersburg	1 Şubat 2019	Constantinople güncellemesi, işlem maliyetlerini düşürmeyi ve ağ verimliliğini artırmayı hedefledi. Ancak güvenlik endişeleri nedeniyle St. Petersburg güncellemesiyle bazı değişiklikler geri alındı.	Hayır
Istanbul	1 Aralık 2019	Ağ performansını artırdı, gaz ücretlerini optimize etti ve zk-SNARKs ile zk-STARKs gibi sıfır bilgi kanıtlarına daha fazla destek ekledi.	Hayır
Muir Glacier	1 Ocak 2020	Zorluk bombasını erteleyerek madencilik zorluğunun ani artışını önledi ve ağın istikrarlı çalışmasını sürdürdü.	Hayır
Berlin	1 Nisan 2021	Gaz ücretlerinde iyileştirmeler yaptı, yeni işlem türleri ekledi ve Ethereum Sanal Makinesi'nde (EVM) çeşitli optimizasyonlar gerçekleştirdi.	Hayır
London	1 Ağustos 2021	EIP-1559'u tanıttı; bu, işlem ücretlerinin bir kısmının yakılmasını sağlayarak ETH arzını azaltmayı ve kullanıcıların öngörülebilir işlem ücretleri ödemesini hedefledi.	Hayır
Altair	1 Ekim 2021	Beacon Chain için ilk büyük güncelleme. PoS mekanizmasında ekonomik teşvikleri ve cezaları güncelledi, doğrulayıcıların performansını artırdı ve ağın güvenliğini güçlendirdi.	Hayır
The Merge	15 Eylül 2022	Ethereum'un Proof of Work (PoW) konsensüs mekanizmasından Proof of Stake (PoS)'a geçişini tamamladı. Bu, enerji tüketiminde büyük bir düşüş sağladı ve Ethereum 2.0'a geçişin önemli bir adımı oldu.	Hayır
Shanghai / Shapella	12 Nisan 2023	Stake edilmiş ETH'lerin çekilmesine izin verdi, böylece doğrulayıcılar yatırdıkları ETH'leri geri alabilir hale geldi. Ayrıca, ağın verimliliğini artıran ve işlem ücretlerini optimize eden çeşitli iyileştirmeler içerdi.	Hayır
Cancun-Deneb (Dencun)	13 Mart 2024	Proto-Danksharding (EIP-4844) özelliğini tanıttı, bu da büyük veri kümeleri için yeni bir işlem türü oluşturarak blok zincirinin ölçeklenebilirliğini artırdı ve özellikle Rollup tabanlı Layer 2 çözümleri için işlem ücretlerini düşürdü.	Hayır
Pectra	1 Nisan 2025	Ethereum'un planlanan bir sonraki büyük güncellemesi. Akıllı hesaplar (EIP-7702) gibi önemli iyileştirmeler getirmesi bekleniyor. Bu güncellenmenin, kullanıcı deneyimini ve ağ kapasitesini önemli ölçüde artırması hedefleniyor.	Hayır

Ethereum oldukça esnek bir yapıya sahip kurgulanmıştır. Ethereum'un temel teknik kuralları ve protokolü Yellowpaper adlı teknik dokümanda belirtilmektedir. En güncel yellowpaper Berlin versiyonudur [63]. Yellowpaper, Ethereum ağının kurallarını, matematiksel modellerini ve teknik özelliklerini net biçimde ortaya koymaktadır ve bu kurallara uygun olarak geliştirilen herhangi bir yazılım, teorik olarak Ethereum ağı üzerinde istemci (client) olabilir. Aşağıda aktif olarak kullanılan istemci listesi verilmiştir. Daha önceden oluşturulmuş ancak artık aktif olmayan birçok istemci bulunmaktadır. Ancak Ethereum resmi olarak Go diliyle yazılmış olan Geth istemcisini desteklemektedir. <https://geth.ethereum.org> adresinden bu istemci indirilebilmektedir.

Tablo 9: Ethereum istemcileri

İstemci	Dil	Durum	Açıklama
Geth	Go	Aktif	En yaygın kullanılan istemci. Ethereum Foundation tarafından destekleniyor.
Nethermind	C# / .NET	Aktif	Windows uyumlu, hafif ve hızlı istemci.
Besu	Java	Aktif	Hyperledger projesinin parçası. Kurumsal kullanım için tasarlanmıştır.
Erigon	Go (+Rust)	Aktif	Arşiv node'lar için optimize edilmiş, çok hızlı istemci.
Reth	Rust	Yeni/deneysel	Flashbots tarafından geliştiriliyor. Modern ve modüler yapıya sahip.
Lodestar	TypeScript	Aktif	Genellikle eğitim ve araştırma amaçlı, daha çok Beacon Chain istemcisi olarak bilinir.
Nimbus	Nim	Aktif	Düşük kaynak tüketimiyle öne çıkar. Hem execution hem consensus istemcisi mevcut.
Prysm	Go	Aktif	Beacon Chain (PoS consensus) istemcisidir.
Lighthouse	Rust	Aktif	Popüler Beacon Chain istemcisi.

Ethereum 2022 yılında Merge hard forkuna kadar PoW konsensüs mekanizması kullanmış ve çalışma mekanizması Bitcoin benzeri devam etmiştir. Bu hard fork sürecinden sonra PoS konsensüs mekanizmasına geçilmiş ve PoW terk edilmiştir. Ancak blokların başlıklarının hash değeri alınarak bir sonraki bloğa yazılma süreci iki coinde de aynı şekilde devam etmektedir. Aşağıda Ethereum kaynak kodundan alınmış başlık birleşenlerini gösteren kod parçası yer almaktadır. Bakıldığında Ethereum'un Bitcoin'e göre çok daha fazla başlık parametresi içerdiği ve daha karmaşık bir mimari kurguladığı görülmektedir. Ethereum üzerinden sadece para transferi yapılmadığı, akıllı sözleşmeler çalıştığı için bu sözleşmelerde harcanacak "Gas" miktarı ile ilgili bilgiler başlıkta yer almaktadır. PoW

zamanından kalma Diffuculty, MixDigest, Nonce gibi değerler şu an anlamsız olsa da geçmiş data ile bağıın kopmaması adına başlıkta yer almaya devam etmektedir.

Ethereum Bitcoin'e göre daha karmaşık bir yapıya sahiptir. Merge hard forkuna kadar tek katmanlı bir zincir olarak çalışırken PoS'e geçiş sonrası 2 katmanlı bir zincire dönüşmüştür. Sonradan gelen Bacon Chain katmanı PoS doğrulayıcılarının organize edildiği ve blokların zamanlamasının yapıldığı yeni zincirdir. Eskiden Bitcoin gibi zorluk ayarlamaları ile 14 sn de bir blok üretimi hedeflenirken artık bu süre 12 saniye olarak sabitlenmiştir. Bu yeni zincir sayesinde madenciye ihtiyaç olmadan stake miktarına göre rastlantısal seçilen doğrulayıcılar blok üretimini gerçekleştirmektedir. Üretilen bu bloklar daha sonra asıl zincire (Execution Layer) geçmektedir.

Kod 19: Ethereum blok başlıklarını gösteren kod parçası

```
go Kopyala  
  
// Header represents a block header in the Ethereum blockchain.  
type Header struct {  
    ParentHash common.Hash    `json:"parentHash"      gencodec:"required"`  
    UncleHash  common.Hash    `json:"sha3Uncles"      gencodec:"required"`  
    Coinbase   common.Address `json:"miner"           gencodec:"required"`  
    Root       common.Hash    `json:"stateRoot"       gencodec:"required"`  
    TxHash     common.Hash    `json:"transactionsRoot" gencodec:"required"`  
    ReceiptHash common.Hash    `json:"receiptsRoot"    gencodec:"required"`  
    Bloom      Bloom          `json:"logsBloom"       gencodec:"required"`  
    Difficulty *big.Int      `json:"difficulty"      gencodec:"required"`  
    Number     *big.Int      `json:"number"          gencodec:"required"`  
    GasLimit   uint64         `json:"gasLimit"        gencodec:"required"`  
    GasUsed    uint64         `json:"gasUsed"         gencodec:"required"`  
    Time       uint64         `json:"timestamp"       gencodec:"required"`  
    Extra      []byte         `json:"extraData"       gencodec:"required"`  
    MixDigest  common.Hash    `json:"mixHash"         gencodec:"required"`  
    Nonce      BlockNonce     `json:"nonce"           gencodec:"required"`  
    BaseFee    *big.Int      `json:"baseFeePerGas"    rlp:"optional"`  
}
```

Ethereum daha karmaşık bir yapıya sahip olmasına karşın adreslerin üretimi konusunda Bitcoin'e göre daha kullanışlı bir kurgu yapmıştır. Bitcoin'de karakter uzunluğu 26-35 karakter olabilirken bu Ethereum da 42 karakter olarak sabittir. Bitcoin de farklı adresleri ifade eden 1,3 veya bc1 ile başlayan adresler bulunurken Ethereum da bütün adresler 0x ile başlamaktadır. Ayrıca Ethereum'da içerik olarak farklı iki adres tipi bulunmaktadır. Bunlardan biri Bitcoin adreslerine karşılık gelen EOA (Externally Owned Account) diğeri ise akıllı sözleşme adresleridir ve tamamen aynı formatta üretilmektedirler. Adres üretiminde iki zincirde ECDSA (Elliptic Curve Digital Signature Algorithm) ve aynı eğri standardı secp256k1 kullanılmaktadır. Ancak hash algoritması olarak Bitcoin SHA-256 ve RIPEMD-160 kullanırken Ethereum Keccak-256 hash algoritmasını kullanmaktadır.

Burada adresler konusundaki en önemli husus bundan sonra yapacağımız birçok karşılaştırmada da benzer şekilde karşımıza çıkacağı gibi Bitcoin'in bir ödeme sistemi olarak kurgulanırken Ethereum'un bir altyapı sistemi olarak kurgulanmasıdır. Bu sebeple bir Bitcoin adresi sadece BTC tutabilirken Ethereum adresi ETHER dışında Ethereum standardı (ERC-20, ERC-721(NFT), ERC-1155) ile üretilmiş yüzlerce tokeni yönetebilir.

Ethereum'da işlem süreçleri Bitcoin'den farklı işlemektedir. Öncelikle bir işlemin içerdiği bilgiler oldukça farklıdır. Aşağıda kaynak kodda (Geth v.1.15.7) bu veriler görülebilmektedir.

Kod 20: Ethereum'da bir Legacy (Eski Tip) işlemin parametreleri

```
go Kopyala  
  
type LegacyTx struct {  
    Nonce      uint64      `json:"nonce"   gencodec:"required"`  
    GasPrice   *big.Int    `json:"gasPrice gencodec:"required"`  
    Gas        uint64      `json:"gas"     gencodec:"required"`  
    To         *common.Address `json:"to"      rlp:"nil"`  
    Value      *big.Int    `json:"value"   gencodec:"required"`  
    Data       []byte      `json:"input"   gencodec:"required"`  
    V, R, S    *big.Int    `json:"v"       gencodec:"required"`  
}
```

Kod: core/types/transaction.go

Londra güncellemesiyle birlikte tanıtılan EIP-1559 işlemleri, dinamik ücret yapısını destekler ve maxFeePerGas ile maxPriorityFeePerGas parametrelerini içerir. Bu tür işlemler aşağıdaki yapı ile tanımlanır:

Kod 21: Ethereum’da bir EIP-1559 işlemin parametreleri

```
go Kopyala  
  
type DynamicFeeTx struct {  
    ChainID    *big.Int    `json:"chainId"      gencodec:"required"`  
    Nonce      uint64      `json:"nonce"        gencodec:"required"`  
    GasTipCap  *big.Int    `json:"maxPriorityFeePerGas" gencodec:"required"`  
    GasFeeCap  *big.Int    `json:"maxFeePerGas"  gencodec:"required"`  
    Gas        uint64      `json:"gas"          gencodec:"required"`  
    To         *common.Address `json:"to"           rlp:"nil"`  
    Value      *big.Int    `json:"value"        gencodec:"required"`  
    Data       []byte      `json:"input"        gencodec:"required"`  
    AccessList AccessList  `json:"accessList"   gencodec:"required"`  
    V, R, S    *big.Int    `json:"v"            gencodec:"required"`  
}
```

EIP-2930 ile tanıtılan ve belirli durumlarda gaz maliyetlerini optimize etmek için kullanılan Access List işlemleri, aşağıdaki yapı ile ifade edilir:

Kod 22: Ethereum’da bir EIP-2930 AccessListTx işlemin parametreleri

```
go Kopyala  
  
type AccessListTx struct {  
    ChainID    *big.Int    `json:"chainId"      gencodec:"required"`  
    Nonce      uint64      `json:"nonce"        gencodec:"required"`  
    GasPrice   *big.Int    `json:"gasPrice"     gencodec:"required"`  
    Gas        uint64      `json:"gas"          gencodec:"required"`  
    To         *common.Address `json:"to"           rlp:"nil"`  
    Value      *big.Int    `json:"value"        gencodec:"required"`  
    Data       []byte      `json:"input"        gencodec:"required"`  
    AccessList AccessList  `json:"accessList"   gencodec:"required"`  
    V, R, S    *big.Int    `json:"v"            gencodec:"required"`  
}
```

Konu üzerine ilerlemeden burada bazen süreçlerin ne kadar karmaşıklāşabildiğini göstermek için güzel bir örnek oluřtuđu kanaatindeyiz. Burada paylařtığımız bir iřlemin ierdiği bilgiler kod parası Ethereum’un en gncel srmnde 3 farklı řekilde kullanılabilmektedir. Zamanla gelen gncellemeler koda yeni parametreler eklerken, sistemin mimarisi geređi eski akıllı szleřmelerin ve veri yapılarının da kullanılabilmesi iin bu gncellemeler sonrası btn eski versiyonların canlı tutulması gerekmektedir. Bu durum zamanla olduka karmařık bir yapıya dnřme potansiyeline sahiptir.

řimdi bu parametreleri inceleyelim:

- **Nonce:** Gnderen hesabın, o ana kadar gerekleřtirdiđi iřlem sayısını belirtir.
- **GasPrice:** İřlemde denecek birim gaz bařına cret miktarını tanımlar (sadece legacy iřlemlerde kullanılır).
- **GasTipCap (maxPriorityFeePerGas):** Madencilere ncelik iin verilen bahřiř miktarını belirtir (EIP-1559 iřlemlerinde).
- **GasFeeCap (maxFeePerGas):** Kullanıcının demek istediđi maksimum gaz fiyatını ifade eder (EIP-1559 iřlemlerinde).
- **Gas:** İřlemin gerekleřtirilmesi iin harcanabilecek maksimum gaz miktarını belirtir.
- **To:** İřlemin alıcısının adresidir. Yeni bir kontrat oluřturuluyorsa bu alan boř bırakılır.
- **Value:** İřlemlle birlikte transfer edilen Ether miktarını ifade eder.
- **Data (Input):** İřlemlle birlikte gnderilen ek veriyi ierir. zellikle akıllı kontrat ađrılarında kullanılır.
- **AccessList:** İřlem sırasında eriřilecek depolama alanlarını belirtir ve gaz maliyetlerini optimize etmek iin kullanılır (EIP-2930 iřlemlerinde).
- **V, R, S:** İřlemin imzasını oluřturan bileřenlerdir ve iřlemin btnlđn ve kimliđini dođrulamak iin kullanılır.

V, R, S deđerlerinden gndericinin adres bilgisi elde edilmektedir.

Ethereum blokzinciri, ađ zerindeki her iřlemin belirli bir hesaplama kaynađı tkettiđini kabul eden bir mimari yapıya sahiptir. Bu bađlamda, her iřlem iin bir “gas” (yakıt) bedeli belirlenmiř olup, bu bedel ađın gvenliđini ve iřlem dođruluđunu korumak iin ekonomik bir teřvik ve sınırlama mekanizması iřlevi grr. Gas, aslında Ethereum Sanal Makinesi’nin (EVM) gerekleřtirdiđi her hesaplama adımı iin denen kk bir maliyeti temsil eder.

Örneğin basit bir ETH transfer işlemi ile bir akıllı sözleşme çağrısı arasında ciddi gas farkı olabilir. Çünkü EVM, sözleşme çağrılarında daha fazla hesaplama ve veri depolama (storage) işlemi yapar. Bu da daha yüksek gas tüketimine neden olur. Gas maliyetleri, işlem sırasında belirlenen gas limit ve gas fiyatı (gas price) üzerinden hesaplanır. Gas limit, işlemin ne kadar hesaplama yapılabileceğini belirlerken; gas fiyatı, birim gas başına ödenmek istenen ETH miktarını ifade eder.

Ethereum 2021 yılında yürürlüğe giren EIP-1559 güncellemesi ile birlikte önemli bir değişiklik yapmış ve her işlemde ödenen temel gas ücretinin (base fee) yakılmasını (burning) zorunlu hale getirmiştir. Bu düzenleme, Ethereum'un parasal politikasında deflasyonist bir mekanizma oluşturmayı amaçlamaktadır. Yani, her işlem sonucunda az miktarda ETH geri dönüşü olmayacak şekilde sistemden silinmekte, bu da Ether arzının zamanla azalmasına neden olmaktadır.

Akıllı sözleşmelerin devreye girmesi ile birlikte işlemlerin gas hesaplamaları daha karmaşık bir hale gelmiştir. Çünkü sözleşmeler içerisinde farklı opcode'lar (EVM komutları) farklı gas miktarları tüketmekte ve bazı sözleşmelerin kendi içinde başka sözleşmeleri çağırması gibi durumlar ortaya çıkmaktadır. Bu nedenle Ethereum geliştiricileri, zaman içinde yapılan güncellemelerle opcode'ların gas maliyetlerinde değişikliklere gitmiş, işlem optimizasyonu ve ağ verimliliği sağlamak adına çeşitli düzenlemeler (örneğin EIP-2929, EIP-3529) hayata geçirilmiştir.

Sonuç olarak, gas kavramı yalnızca işlem ücreti değil; Ethereum'un hesaplama ekonomisini, ağ tıkanıklığı kontrolünü, ve deflasyonist ETH arz politikasını birlikte yöneten temel yapı taşlarından biridir. Yukarıdaki kod parçalarında gördüğümüz farklı işlem parametreleri de bu hesaplamaların optimize edilmesi için yapılan güncellemelerden kaynaklanmaktadır. "Gas" hesaplamasını basit birkaç örnekle daha anlaşılır hale geleceği kanaatindeyiz.

- Basit bir ETH transferini ele alalım Legacy İşlem dediğimiz güncelleme öncesi yöntemi kullanırsak hesaplama şöyle olmaktadır:

Standart bir para gönderme işlemi sistem tarafından 21.000 Gas olarak belirlenmiştir. Gas fiyatı ağın yoğunluğuna göre değişmektedir. Gas price: 50 Gwei olduğunu varsayarsak

$$\text{Toplam Ücret} = 21.000 \times 50 \text{ Gwei} = 1.050.000 \text{ Gwei} = 0.00105 \text{ ETH}$$

Bu ücret tamamen madenciye gider ETH yakılmaz. Burada Gas price tamamen kullanıcı tarafından belirlenir. Kullanıcı bunu düşük tutarsa işleminin onaylanması gecikirken yüksek tutarsa zarar etmiş olmaktadır.

- Aynı işlemin EIP-2930 güncellemesi ile gelen yöntemle yapalım. Access List kullanmazsak tamamen Legacy ile aynı olacaktır.
- Şimdi de EIP-1559 güncellemesi ile gelen yöntemle yapalım. Para gönderme işlemi standart olarak 21.000 Gas'dır. Base Fee: 30 Gwei ve Priority Fee(tip): 2 Gwei olsun. Burada Base Fee sistemin yoğunluğu tarafından belirlenen bir değerdir ve sistem tarafından otomatik belirlenir. Priority Fee ise kullanıcı tarafından belirlenir. Böylelikle kullanıcının Gas Price hesaplarken zorlanmasının önüne geçilmiştir. Toplam ücreti şimdi hesaplayalım.

$$\text{Yakılan ETH} = 21.000 \times 30 \text{ Gwei} = 630.000 \text{ Gwei} = 0.00063 \text{ ETH}$$

$$\text{Madenciye Giden ETH} = 21.000 \times 2 \text{ Gwei} = 42.000 \text{ Gwei} = 0.000042 \text{ ETH}$$

$$\text{Toplam Ücret} = 0,000672 \text{ ETH}$$

İşlem daha ucuza gerçekleşmekle birlikte harcanan ETH'ın büyük kısmı yakılarak (burn) arz azaltıcı bir sonuç oluşmaktadır.

Şimdi bir akıllı sözleşme çalıştırma örneği ile tekrar hesaplama yapalım. ERC-20 üzerinden bir token transferi işlemi olsun. Örneğin 100 LINK gönderelim.

- Legacy İşlem olarak gerçekleşirse Gas used: 50.000 , Gas Price: 60 Gwei olsun
$$\text{Toplam Ücret} = 50.000 \times 60 \text{ Gwei} = 3.000.000 \text{ Gwei} = 0.003 \text{ ETH}$$

Tamamı madenciye gidecektir.

- Aynı işlemin EIP-2930 güncellemesi ile gelen yöntemle yapalım. (1 adres, 1 storage Access List ile)

Base gas: 50.000, Access list: 1 adres (2400 gas) + 1 storage key (1900 gas)

Toplam gas: 54.300 Gas price 60 Gwei

Toplam Ücret = $54.300 \times 60 \text{ Gwei} = 3.258.000 \text{ Gwei} = 0.003258 \text{ ETH}$

Tamamı madenciye gidecektir.

- Şimdi de EIP-1559 güncellemesi ile gelen yöntemle yapalım.

Gas used:50.000, Base Fee:40 Gwei, Priority Fee: 3 Gwei

Yakılan ETH = $50.000 \times 40 \text{ Gwei} = 2.000.000 \text{ Gwei} = 0.002 \text{ ETH}$

Madenciye Giden ETH = $50.000 \times 3 \text{ Gwei} = 150.000 \text{ Gwei} = 0.00015 \text{ ETH}$

Toplam Ücret = 0,00215 ETH

Tablo 10: Ethereum Yellow Paper Appendix G'den işlem ve gas maliyetleri listesi

	İşlem (Opcode veya İşlem Tipi)	Gas Maliyeti
1	Transaction (Base Cost)	21,000 gas
2	Contract Creation (Base Cost)	32,000 gas
3	SSTORE (Storage Write New Value)	20,000 gas (first write)
4	SSTORE (Update to Same Value)	5,000 gas (update existing)
5	SLOAD (Storage Read)	100 gas
6	CALL (External Contract Call)	700 gas + call cost
7	BALANCE (Check Address Balance)	400 gas
8	EXTCODESIZE (Size of External Code)	700 gas
9	EXTCODECOPY (Copy External Code)	700 gas + copying cost
10	SHA3 (Keccak256 Hash)	30 gas + 6 gas/word
11	LOG (Create Event)	375 gas + 8 gas/byte
12	EXP (Exponentiation)	10 gas + 10 gas/byte
13	Self-Destruct (SUICIDE)	5,000 gas (can refund)
14	Memory Expansion (per word)	3 gas per word (32 bytes)
15	CALLDATACOPY (Copy Input Data)	3 gas per word (copy cost)

Ethereum'da bir işlem kullanıcının kendi cüzdanı üzerinden oluşturulur. Oluşturulan işlem öncelikle kullanıcının bağlı olduğu düğüme ulaşır. Daha sonra bu işlem, diğer düğümlere hızlıca yayılır ve tüm Ethereum tam düğümlerinin ortak kullandığı genel bir işlem havuzu (Transaction Pool ya da Mempool) içerisinde beklemeye alınır. Bu havuz, henüz bir bloğa

eklenmemiş, onaylanmayı bekleyen tüm işlemlerin biriktiği ortak ve dağıtık bir yapıdır. Ethereum'da PoS yapısında blok üretme yetkisi, ağ tarafından stake edilen Ether miktarıyla doğru orantılı rastgele seçilen validatöre verilir. Seçilen validatör, işlem havuzundan işlemleri belirli kurallara göre seçip bloğa yerleştirir ve oluşturduğu blokta yer alacak işlemleri belirleyip kendi Ethereum Sanal Makinesi'nde (EVM) çalıştırır. Bu aşamada, eğer işlem bir akıllı sözleşme çağırıyorsa, akıllı sözleşmenin kodu EVM tarafından çalıştırılır ve sözleşmenin içindeki tüm hesaplamalar yapılır. Eğer sözleşmenin çalıştırılması sonucunda Ether transferleri veya durum değişiklikleri varsa, bu bilgiler blok içine yazılır. Validatör, işlemler sonucu ortaya çıkan tüm durum değişikliklerini içeren bloğu oluşturur ve blok bilgilerini kriptografik olarak imzalar. Validatör, imzaladığı bloğu Ethereum ağındaki diğer tam düğümlere dağıtır (broadcast). Bu yeni bloğu alan düğümler kendi Ethereum Sanal Makineleri'nde (EVM) aynı işlemleri tekrar çalıştırarak blokta yer alan hesaplamaların ve durum değişikliklerinin doğruluğunu kontrol eder. Ağdaki diğer tam düğümler, bloktaki işlemleri ve durumları doğruladıktan sonra blok zincirine bu bloğu ekler ve üzerinde fikir birliği (konsensüs) oluşur. Bu işlem sonucunda Ethereum blok zincirinde yeni durum ortaya çıkar ve işlemler kalıcı hale gelir.

Burada Ethereum sanal makinesi (EVM- Ethereum Virtual Machine) konusunda değinmenin faydalı olacağı kanaatindeyiz. EVM Ethereum ağı üzerindeki tüm düğümlerde çalışan ve akıllı sözleşmelerin yürütüldüğü evrensel, deterministik bir sanal bilgisayardır. Her türlü algoritmayı çalıştırabilecek yani Turing-complete bir sanal makinedir. Deterministiktir, yani her düğüm, aynı işlemleri aynı sözleşmeleri çalıştırdığında her zaman aynı sonucu alır. Yığın tabanlıdır. Gas sistemi ile çalışır. Yapılan her işlemin bir gas maliyeti vardır. Böylelikle sonsuz döngüler engellenir. EVM'nin nasıl çalıştığını anlamak için bir akıllı sözleşme örneği incelemenin daha verimli olacağı kanaatindeyiz.

Ethereum da sözleşmelerin %90'dan fazlası Solidity ile yazılmaktadır. Solidity, Ethereum için özel olarak geliştirilmiş yüksek seviyeli, nesne yönelimli bir programlama dilidir. Ethereum Virtual Machine (EVM) ile uyumludur. C/C++ ve JavaScript'e benzer sözdizimi kurallarına sahip olduğu için öğrenmesi görece kolaydır. Özellikle akıllı sözleşmelerin mantığını ve güvenliğini ele alacak şekilde tasarlanmıştır. Aşağıda <https://docs.soliditylang.org/en/latest/solidity-by-example.html> sitesinden alınmış bir akıllı

sözleşme örneği yer almaktadır. Bu sözleşmeyi incelemenin konunun anlaşılması yönünde oldukça etkili olacağı kanaatindeyiz.

Tablo 11: Solidity ile yazılmış bir akıllı sözleşmenin incelenmesi

	Solidity Sözleşme Kodu	Açıklama
1	// SPDX-License-Identifier: GPL-3.0	Serbet lisansa sahip olduğunu gösteren bir ifade
2	pragma solidity >=0.7.0 <0.9.0;	Sözleşmenin Solidity'nin hangi versiyonları ile derlenebileceğini gösteren ifade
3	/// @title Vekâletle Oy Kullanma Sözleşmesi	
4	contract Ballot {	Ballot adında bir sözleşme oluşturuluyor.
5	// Seçmenleri temsil eden veri yapısı	
6	struct Voter {	Seçim yapacak kişiler için oluşturulan değişken 4 özelliğe sahip. Oy kullanma ağırlığı. Başlangıçta bir eğer başka oy verecekler tarafından vekalet verilirse vekalet sayısı kadar artacak. Oy kullanıp kullanmadığı. Oy hakkını devrettiği kişi varsa onun adresi verdiği oyun oy verilecekler dizindeki yeri.
7	uint weight; // Oy hakkı ağırlığı (başlangıçta 1 olabilir, delegasyonla artar)	
8	bool voted; // Bu kişi oy kullandı mı?	
9	address delegate; // Oy hakkını devrettiği kişi (varsa)	
10	uint vote; // Oy verdiği teklifin dizindeki index'i	
11	}	
12	// Teklifleri temsil eden veri yapısı	
13	struct Proposal {	Oylama yapılacak konuyu temsil etmek için kullanılacak değişken. İki parametresi var teklifin ismi ve teklifin aldığı oy sayısı
14	bytes32 name; // Teklifin kısa ismi (32 byte'a kadar)	
15	uint voteCount; // Bu teklifin aldığı toplam oy sayısı	
16	}	
17	address public chairperson; // Başkan: oy hakkı verme yetkisi olan kişi	Bir başkan oluşturuldu.
18	mapping(address => Voter) public voters; // Adresleri seçmen bilgileriyle eşleyen harita	adrese karşılık bir seçmen oluşturulacak yapı tanımlandı.
19	Proposal[] public proposals; // Tüm teklifler bu dinamik dizide tutulur	Teklifleri tutmak için bir dizi yapısı oluşturulur.
20	/// Sözleşme kurulduğunda çalışır. Teklif isimlerini alır ve başkanı tanımlar.	Burada sözleşme ilk kurulduğunda yapılandırılır. Sözleşmeyi sisteme yükleyen kişi başkan olmaktadır. Başkan tanımlandıktan sonra hemen oy hakkıda tanımlanır. Başkanın yolladığı seçim yapılacak isimler proposal değişkenine eklenir.
21	constructor(bytes32[] memory proposalNames) {	
22	chairperson = msg.sender; // Sözleşmeyi oluşturan kişi başkan olur	
23	voters[chairperson].weight = 1; // Başkanın oy hakkı başlangıçta 1'dir	
24	// Gelen teklif isimleri tek tek proposals dizisine eklenir	
25	for (uint i = 0; i < proposalNames.length; i++) {	
26	proposals.push(Proposal({	
27	name: proposalNames[i],	
28	voteCount: 0	

29	});	
30	}	
31	}	
32	/// Oy hakkı verme fonksiyonu. Sadece başkan çağırabilir.	
33	function giveRightToVote(address voter) external {	Burada tanımlanan oyhakkıverme fonksiyonu sadece dışardan başkan tarafından çağırılabilir. 35 satır oy kullanacak kişinin daha önceden kullanıp kullanmadığını kontrol eder. 36 satır kişiye 2 kere oy hakkı verilmesini engeller. Aksi durumlarda hata mesajı verilmektedir. Herhangi bir hata oluşmazsa ilgili adrese bir oy hakkı verilir.
34	require(msg.sender == chairperson, "Sadece başkan oy hakkı verebilir.");	
35	require(!voters[voter].voted, "Bu kisi zaten oy kullandı.");	
36	require(voters[voter].weight == 0, "Bu kisiye daha once oy hakkı verilmiş.");	
37	voters[voter].weight = 1; // Oy hakkı tanımlanır	
38	}	
39	/// Oy hakkını başka birine devretme fonksiyonu	
40	function delegate(address to) external {	Burada seçime katılmak istemeyenlerin bir vekalet atama temsilci belirleme süreci işletilmektedir. Bu fonksiyon da sözleşme dışından çağrılan bir fonksiyondur. 42., 43. ve 44. satırlarda sırasıyla temsilci belirlemek için fonksiyonu çağırان kişinin oy hakkı olup olmadığı, daha önce oy kullanıp kullanmadığı ve kendini temsilci yapmaya çalışıp çalışmadığı tespit edilir. Bu kontrollere takılmadıysa kod çalışmaya devam eder. 46. satırda while döngüsü ile temsilci seçilen kişinin de kendine temsilci seçip seçmediği ve bu zincirin devamı bulunmaktadır. Ancak temsilci seçmek için bu fonksiyonu çağırان kişinin bu zincirde bulunup döngü yaratması 48. satırla engellenir. Temsilci zincirinde en son kişi bulunup onun için kontroller yapılır. Temsilci seçilecek kişinin oy hakkı var mı 51. satır ile kontrol edilir. Varsa işlem devam eder ve 52. ve 53. satırlarla bu fonksiyonu çağırان kişi oy kullanmış ve temsilci seçmiş olarak sisteme kaydedilir. 54. satır ile temsilci oy kullanmışsa onun seçtiği teklife oy eklenir. Henüz kullanmamışsa 59. satırla oy hakkı artırılır.
41	Voter storage sender = voters[msg.sender]; // Gönderenin bilgilerini al	
42	require(sender.weight != 0, "Oy hakkın yok.");	
43	require(!sender.voted, "Zaten oy kullandın.");	
44	require(to != msg.sender, "Kendine delege edemezsin.");	
45	// Delege edilen kişi de başka birine delege ettiyse zinciri takip et	
46	while (voters[to].delegate != address(0)) {	
47	to = voters[to].delegate;	
48	require(to != msg.sender, "Delegasyon döngüsü olusturulamaz.");	
49	}	
50	Voter storage delegate_ = voters[to];	
51	require(delegate_.weight >= 1, "Delege edilen kisinin oy hakkı yok.");	
52	sender.voted = true;	
53	sender.delegate = to;	
54	if (delegate_.voted) {	
55	// Delege edilen kişi zaten oy kullanmışsa, oyunu doğrudan onun seçtiği teklife ekle	
56	proposals[delegate_.vote].voteCount += sender.weight;	
57	} else {	
58	// Aksi halde, onun oy hakkına ekle	
59	delegate_.weight += sender.weight;	
60	}	
61	}	
62	/// Oy verme fonksiyonu	Bu fonksiyonda sözleşme dışından çağrılmaktadır. Oy vermek isteyen kişi tercihini parametre olarak gönderir. 65. satır ile oy kullanma hakkı olmayan kullanıcılar elenmektedir. 66. satır daha önce oy kullanan kullanıcıları eler. 67. satırla oyun kullanıldığı kayıt altına alınır.
63	function vote(uint proposal) external {	
64	Voter storage sender = voters[msg.sender];	
65	require(sender.weight != 0, "Oy hakkın yok.");	
66	require(!sender.voted, "Zaten oy kullandın.");	
67	sender.voted = true;	

68	sender.vote = proposal;	68. satırda oy verilecek indeks bilgisi alınır ve 69. satırda bu indeksteki seçime oy kullananın oy ağırlığı kadar oy eklenir.
69	proposals[proposal].voteCount += sender.weight; // Oy ağırlığı kadar oy eklenir	
70	}	
71	/// En çok oy alan teklifin index'ini hesaplar	Bu fonksiyon herkes tarafından çağrılabilir. Blok zincire veri yazan değil sadece okuma yapan bir fonksiyondur. Gas harcamaz. 74. satırda bütün tekliflerin karşılaştırılarak en fazla oy almış teklifin winningVoteCount değişkenine bu teklifin indeksinin de winningProposal_ değişkenine atanması sağlanır.
72	function winningProposal() public view returns (uint winningProposal_) {	
73	uint winningVoteCount = 0;	
74	for (uint p = 0; p < proposals.length; p++) {	
75	if (proposals[p].voteCount > winningVoteCount) {	
76	winningVoteCount = proposals[p].voteCount;	
77	winningProposal_ = p;	
78	}	
79	}	
80	}	
81	/// Kazanan teklifin adını döner	sözleşme dışından çağrılan ve sadece okuma yapan bir fonksiyondur. 72. satırdaki wininPrposel() fonksiyonunu çağırır. Bu fonksiyon kazananın indeksini dönderir ve kazananın ismi 82. satırdaki fonksiyonla ilan edilmiş olur.
82	function winnerName() external view returns (bytes32 winnerName_) {	
83	winnerName_ = proposals[winningProposal()].name;	
84	}	
85	}	

Konunun ayrıntılarının daha iyi anlaşılması için bu sözleşmeyi kullanacağımız bir örnek aşağıda yer almaktadır.

Örnek: Ali bu yıl doğum gününü sıradan bir kutlamayla geçirmek istemiyordu. Arkadaşlarıyla beraber karar vereceği ama aynı zamanda eğlenceli ve yenilikçi bir yöntem kullanmak istiyordu. Kripto dünyasına olan ilgisi sayesinde aklına ilginç bir fikir geldi: Neden Ethereum üzerinde bir akıllı sözleşmeyle oylama yapmalıyım. Hem teknolojik bir yaklaşım olurdu hem kimse alınmazdı, hem de herkesin tercihi şeffaf şekilde kaydedilirdi. Böylece doğum günü planı demokratik bir şekilde belirlenmiş olurdu. Ali, doğum günü için dört seçenek belirledi: evde eğlenceli bir parti, dışarıda şık bir mekânda akşam yemeği, canlı müzik yapan bir mekânda gece eğlencesi ve açık havada bir parkta buluşma. Sözleşmeyi kurduktan sonra oy hakkı vereceği arkadaşlarını belirledi. Ayşe, Fatma, Mehmet, Murat, Ömer, Yeliz ve Merve doğum günü planı konusunda karar verecek yedi yakın arkadaşıydı. Ali de oy kullanacak hem organizatör hem seçmen olacaktı. Bazı arkadaşları kendi tercihlerinden çok, güvendikleri kişilere karar hakkını bırakmak istediler. Ömer sevgilisi

Merve'ye güvenerek "Benim adıma Merve karar versin," dedi. Yeliz de Merve'yi çok sevdiği için oy hakkını ona devretti. Fatma ise, "Benim için Mehmet ne derse odur," diyerek oyunu ona bıraktı.

Ali, şık bir mekânda yemek yemeyi tercih etti. Ayşe, evde samimi bir parti fikrine oy verdi. Mehmet müzik tutkunu olduğu için canlı müzik seçeneğini işaretledi. Murat da Mehmet'le aynı fikirdeydi ve canlı müzikten yana oy kullandı. Merve ise akşam yemeğini seçti. Ömer ve Yeliz daha önceden oylarını Merve'ye devrettiği için onların tercihleri de bu yönde sayıldı. Fatma'nın oy ise Mehmet'in seçimini izledi.

Örneğimizde sözleşmeyi kurgulayan Ali olduğu için sözleşmeyi sisteme yüklerek başlatır ve chairperson olur.

Ballot(["Evde parti", "Mekana yemek", "Canli muzik", "Parkta bulusma"])

Böylelikle 4 tane teklif içeren bir proposal oluşturulur.

Ali tüm arkadaşlarına oy hakkı verir:

giveRightToVote(ayse)

giveRightToVote(omer)

giveRightToVote(fatma)

giveRightToVote(yeliz)

giveRightToVote(mehmet)

giveRightToVote(merve)

giveRightToVote(murat)

Ali'nin oy hakkı alan arkadaşları bu haklarını kullanırlar ya da bir temsilci atarlar.

Örneğimize göre aşağıdaki atamalar yapılır.

- *omer.delegate(merve)*
- *yeliz.delegate(merve)*
- *fatma.delegate(mehmet)*

Bu atamalardan sonra Ömer Yeliz ve Fatma artık oy kullanamaz. Oy kullanma işlemi ise aşağıdaki gibi olur. Vote() fonksiyonu proposal'da yer alan tercihlerin indeks koduyla gönderilir.

- *ali.vote(1) // Mekanda yemek*
- *ayse.vote(0) // Evde parti*
- *mehmet.vote(2) // Canlı müzik*
- *murat.vote(2) // Canlı müzik*
- *merve.vote(1) // Mekanda yemek*

Bu oylama sonucunda “Mekanda Yemek” seçilmiş olur çünkü Merve 3 oy hakkına sahiptir ve Ali’nin oyuyla toplam 4 oy ile bu seçim yapılmış olur.

4.3. Geleneksel İşlemlerle Blok Zinciri Teknolojisinin Karşılaştırılması

Blok zinciri teknolojisinin sunduğu yenilikler, sadece teknik boyutuyla değil; sosyal, ekonomik ve hukuki boyutlarıyla da geleneksel sistemlere alternatif oluşturmaktadır. Bu bölümde, tez içerisinde anlattığımız bazı konular kapsamında farklı sektör ve işlemler özelinde blok zinciri uygulamaları ile geleneksel yöntemler karşılaştırmalı olarak ele alınmasının konunun daha iyi anlaşılması için faydalı olacağı kanaatindeyiz.

Bazı bankacılık işlemlerinin Geleneksel Yöntem ya da Bitcoin ile gerçekleştirilmesi:

Geleneksel bankacılık sisteminde, para transferi süreci birçok aracı kurumdan geçer. Özellikle uluslararası transferlerde aracı bankalar (correspondent banks), Swift sistemi, işlem saatleri ve regülasyonlar işlemin süresini ve maliyetini artırmaktadır. Bir kullanıcının yurtdışına para göndermesi günler sürebilmekte ve %2 ila %10 arasında değişen komisyonlar alınabilmektedir. Bitcoin ise bu süreci doğrudan kişiler arası (peer-to-peer) yapısı sayesinde saniyeler ile birkaç dakika arasında gerçekleştirebilmekte, işlemler ağ yoğunluğuna bağlı olarak çok düşük komisyonlarla tamamlanabilmektedir. Bankacılık sisteminde işlem kayıtları yalnızca banka tarafından tutulurken, Bitcoin’de tüm işlemler şeffaf bir şekilde herkesin erişimine açık blok zincirinde saklanmaktadır. Bu durum, hem güvenlik hem de denetlenebilirlik açısından büyük bir fark yaratmaktadır.

Geleneksen bankacılık sisteminde banka müşterisi olmak ve işlem geçmişinizi görmeniz kanunla ve alt düzenlemelerle belirlenmiş kurallara tabidir. Bazı insanların banka müşterisi olması ve işlem yapması hakkında kısıtlar getirilebilir. Bankacılık hizmetlerinden

yararlanması sınırlandırılabilir. Örneğin bankalar risk skorunuza göre size kredi vermeyi reddedebilir. Yurtdışı yerleşik birini müşteri olarak kabul etmeyebilir. Banka müşterisi Bankanın izin verdiği oranda ve sadece kendine ait hesap hareketlerini takip edebilir. Bitcoin de ise herhangi biri kimlik bilgisi vermeden sisteme dahil olabilir. Hiçbir kısıtlamaya tabi olmadan işlem gerçekleştirebilir ve sadece kendisine değil bütün sisteme ait işlem hareketlerine her zaman ulaşabilir. Blok zinciri ekosisteminde yer alan merkeziyetsiz yapılardan kredi kullanabilir.

Burada blok zinciri teknolojisinin geleneksel bankacılıktan çok daha etkili ve efektif olduğu görülmektedir. Ancak yapılan hatalı işlemlerin düzeltilmesi, hırsızlık ya da siber saldırı olaylarına karşı kişi haklarının korunması ile işlemlerin kolay ve anlaşılabilir şekilde gerçekleştirilebilmesi hususunda geleneksel yöntemlerin hala güçlü olduğu görülmektedir.

Akıllı Sözleşmeler ve Geleneksel Hukuki Prosedürler: Geleneksel sözleşme süreçleri, tarafların anlaşması sonrası genellikle noter ya da avukat gibi yasal güvence sağlayan kişiler tarafından yürütülür. Bu süreçler zaman alıcı, maliyetli ve fiziksel varlık gerektiren uygulamalardır. Ayrıca sözleşmenin uygulanması da çoğu zaman bir otoritenin devreye girmesiyle mümkün hale gelmektedir. Akıllı sözleşmeler (smart contracts), blok zinciri üzerinde yazılım diliyle kodlanan, belirli koşullar gerçekleştiğinde otomatik olarak yürürlüğe giren dijital sözleşmelerdir. Bu yapı sayesinde herhangi bir üçüncü tarafa ihtiyaç duymadan işlemler koşullar sağlandığında otomatik olarak gerçekleştirilir. Ethereum gibi platformlar bu yapının en yaygın örneklerini sunmaktadır. Özellikle finans, sigorta, oyun ve tedarik zinciri gibi alanlarda geleneksel sözleşmelerin yerine kullanılmaya başlanmıştır.

Burada en güzel örnek olarak zirai ürünlerin sigortalanması düşünülebilir. Belli alanda belli ürünlerde tabiat olayları kaynaklı zararların gerçekleştiği durumda, bu zararın hava durumu değerleri ile otomatik olarak sözleşmeler tarafından algılanması sonrası üreticinin hesabına para otomatik olarak yatacaktır. Günümüzde ise bu zararın ispatı, sigorta şirketi tarafından durumun kabul edilmesi gibi işlemleri uzatan ve bazen üretici aleyhine sonuçlanan süreçler yaşanmaktadır.

Genel olarak bakıldığında akıllı sözleşmeler her durumda geleneksel sözleşmelerden daha hızlı ve daha verimli sonuç üretmektedir. Ancak burada en büyük sıkıntı akıllı sözleşmelerin nasıl kodlandığı konusunda kullanıcının ayrıntılı bilgiye sahip olabilmesidir. Çünkü yanlış yapılan bir işlemin ya da hatalı kodlanmış bir sözleşmenin geri dönüşü ya da telafisi olmayacaktır.

Blok Zincirinde Araç Kiralama ve Geleneksel Kiralama Şirketleri: Geleneksel araç kiralama sürecinde müşteriler, aracı bir şirketten belirli şartlara göre kiralar. Bu süreçte kimlik doğrulama, sözleşme imzalama, teminat ödeme ve fiziki teslim gibi adımlar yer alır. Sözleşme şartlarının uygulanması ise tamamen şirketin inisiyatifine kalmış durumdadır. Anlaşmazlık durumlarında kullanıcılar genellikle hukuki süreçlere başvurmak zorunda kalmaktadır.

Tezin “Tokenize Önerisi” başlıklı 7. Bölümünde de ayrıntılı olarak ele aldığımız Blok zinciri teknolojisine dayalı olarak geliştirilen araç kiralama sisteminde, her bir araç tokenlaştırılarak yatırımcıların ortak sahipliğine sunulmakta, araçların kiralama işlemleri ise akıllı sözleşmeler aracılığıyla yönetilmektedir. Kullanıcılar, aracı kiralamak istediklerinde kimlik doğrulama, ödeme, zamanlama gibi süreçleri doğrudan blok zinciri üzerinde ve merkeziyetsiz biçimde gerçekleştirmektedir. Kiralama süresi dolduğunda ya da şartlar yerine getirilmediğinde akıllı sözleşme otomatik olarak devreye girer. Bu model sayesinde hem şirketler daha güvenli bir gelir modeli elde eder, hem de kullanıcılar daha şeffaf ve denetlenebilir bir kiralama deneyimi yaşar.

Burada müşterinin blok zincirine bakarak arabanın geçmişi hakkında ayrıntılı bilgi alabildiğini, böylece araç kiralama şirketi tarafından sorunlu ya da bakımları yapılmamış bir araç kiralama imkanının ortadan kalktığını görmekteyiz. Müşterinin kredi kartı bilgilerinin alınması ya da boş kâğıda imza attırılması gibi suiistimal edilebilecek durumlarından da ortadan kaldırılması sağlanmaktadır.

Fiziki Seçim ve Blok Zinciri ile Oylama: Fiziki oylamada seçmenler sandığa gitmek zorundadır. Oy verme, sayım ve sonuç ilanı süreçleri zaman alır. Ayrıca oy manipülasyonu

ve geçersizlik gibi sorunlar yaşanabilir. Blok zinciri ile e-oylama sistemlerinde her bireye dijital cüzdan tanımlanabilir. Oylar zincire kaydedilir, değiştirilemez ve şeffaftır. Estonya bu sistemin öncüsüdür. Ayrıca yurt dışındaki vatandaşların güvenli oy kullanımı da bu yolla sağlanabilir.

Estonyanın son seçimlerde karma bir sistem kullanması ve nerdeyse oy kullananların yarısından fazlasının e-oynamayı tercih etmesinin insanların blok zinciri teknolojisine çok hızlı alışabileceğinin en güzel örneği olduğu kanaatindeyiz. Geçersiz oy kavramının ortadan kalkacak olması, sonuçların çok hızlı açıklanabilmesi, oy çalma gibi durumların söz konusu olmaması ve fiziksel seçim süreci masraflarından kurtulunması gibi birçok olumlu sonuç dikkat çekmektedir.

Blok zinciri uygulamaları, geleneksel sistemlerin birçok eksikliğini gidermeye adaydır. Özellikle araçların ortadan kalkması, işlem sürelerinin kısalması, şeffaflık, maliyetlerin düşmesi ve otomasyon gibi avantajları sayesinde çok çeşitli sektörde dönüşüm yaratmaktadır. Bununla birlikte teknolojinin nispeten yeni olması, regülasyon eksiklikleri ve ölçeklenebilirlik sorunları gibi handikapları da göz ardı edilmemelidir.

Gelecekte, geleneksel sistemlerle blok zincirinin birlikte çalıştığı hibrit modellerin gelişmesi beklenmektedir. Bu nedenle karşılaştırmalı analizlerin yapılması, her iki yöntemin güçlü ve zayıf yönlerinin anlaşılması açısından önemlidir.

5. BLOK ZİNCİRİ TEKNOLOJİSİNİN UYGULAMALARI

Blok zinciri teknolojisinin ilk ortaya çıktığı zaman yarattığı büyük bir devrim algısının zamanla etkisini kaybettiği kanaatindeyiz. Zira o zaman yapılan çalışmalarda ve sunumlarda artık merkeziyetsiz bir dünyanın kapılarının açıldığı ve kısa zamanda para transferlerinin ve finansal enstrümanların bu alana taşınacağı, tapu kayıtları, sağlık sistemleri ve lojistik gibi alanlarda blok zinciri teknolojisinin devrim yapılacağı iddia edilmekteydi. Bitcoin'in hayatımıza girdiği 2009 yılından günümüze kadar geçen 16 yıl içerisinde sürecin hiç de böyle işlemediği görülmektedir. Kripto paralar açık ara blok zinciri teknolojisinin en başarılı uygulamaları olmakla birlikte para transferinden ziyade bir yatırım aracı haline gelmişlerdir. Diğer alanlarda ise büyük ölçekli ve kitlesel benimsenmiş bir uygulama görülmemektedir.

Bu durumu değerlendirirken öncelikle teknolojik gelişmelerin sosyo-politik süreçlerden bağımsız olmadığını vurgulamalıyız. Zira Bitcoin'in duyurulması ABD'de yaşanan finansal bir kriz sonrasında gerçekleşmiştir. Bu kriz sırasında ABD hükümeti bazı büyük kuruluşların batmasını önlemek için halkın parasını kullanmakla suçlanmış, devletlerin para üzerindeki hegemonyası ve para basma gücü tartışmaya açılmıştır. Tam da bu sorunlara bir çözüm üreten Bitcoin hayatımıza girerken, bu inovasyonun alt yapısını oluşturan blok zinciri teknolojisi, otoritelerin yarattığı bütün sorunlardan insanları kurtaracak bir kurtarıcı olarak ortaya çıkmıştır. Aynı şekilde 2020-2022 yılları arasında gerçekleşen global pandemi döneminde artık ulusların dünya meselelerinde kendi başlarına başarılı olamayacağı, küresel bir yaklaşımla ülkeler üstü hareket edilmesi gerektiği vurgulanmaktaydı. Evden çıkamayan insanların artık işlerini, alışverişlerini ve sosyal hayatlarını dijital ortamlarda yaşamaya başlaması bu durumun normalleşmesi sonucunu doğurmuştu. Paranın fiziksel özelliğinin kaybolduğu, sosyal medya ve sosyalleşme uygulamalarının hayatın önemli bir parçası haline geldiği, bir lokasyona bağlı kalmadan dünyanın her yerinden çalışılabileceği gerçekleri insan hayatına girdi. Bu süreçte, artık otoritelerin yetersizliği ve gereksizliği fikri güçlenirken, metaverse, blok zinciri ve Web3 gibi teknolojiler sorunların çözümü olarak karşımıza çıkarılmışlardır. Bu teknolojilerin her biri gerçekten çok önemli inovasyonlar olmakla birlikte hiçbirisi vaad edilen başarıya ulaşamadı. Bunun sebebini iki açıdan ele almamız gerekmektedir. Birinci olarak, sosyo-politik anlamda bakıldığında küreselci bakış açısının hâkim olduğu dünyada ulusal otoritelere birer başkaldırı olan bu teknolojilerin çok fazla

parlatılarak piyasaya sürülmesidir. Bu durum beklentileri oldukça yükseltmiş ancak uygulamalar açısından bakıldığında teknolojiler hayatın içine adapte olamamıştır. İkinci olarak ise ölçeklenebilme, karşı çıkılan otoriteler tarafından yapılacak regülasyon belirsizlikleri, teknoloji yatırımlarının büyük kısmını gerçekleştiren teknoloji devlerinin kendi otoritelerini ortadan kaldıracak web 3.0'a geçme konusundaki isteksizlikleri, kripto paralarda yaşanan ve normal kullanıcılar tarafından anlaşılması oldukça zor olan siber saldırı ve hırsızlık olayları gibi oldukça anlaşılabilir nedenler bu teknolojilerin hayatımıza tam olarak girmelerini engellemiştir. Bu teknolojilerin zaten olması gerektiği hızda hayatımıza girdikleri kanaatindeyiz. İlerde geliştirilecek yeni teknolojiler ile bu süreç hızlanacaktır.

Günümüzde ise sosyo-politik açıdan dünyada yeniden bir ulusallaşma yaklaşımının etkin olduğu görülmektedir. Birçok ülkede aşırı sağ ya da muhafazakâr hükümetler iktidara gelmekte ya da güçlenmektedir. Bu durum ülkelerin kontrolünü devre dışı bırakacak teknolojilerin ya da uygulamaların bir süre gündeme gelmeyeceğini işaret etmektedir. Bunun en güzel örneği merkez bankalarının dijital para geliştirme çabalarıdır. Kripto paralara karşı kendi kuralları ile yönetecekleri ulusal dijital paralar yaratılması planlanmaktadır. Günümüz teknoloji alanında yapılan çalışmalara bakıldığında az önce bahsettiğimiz teknolojilerin popülerliğini kaybettiği ve bütün rekabetin ve yatırımların yapay zekâ alanına kaydığı görülmektedir.

Aslında blok zinciri teknolojisinin asıl patlamasını yapay zekâ ile entegrasyon sonrası yapacağı kanaatindeyiz. Blok zinciri teknolojisinin temel sorunu olarak gördüğümüz merkeziyetsizliğin yarattığı güvenlik ve düzenleme sorunlarının, yapay zekâ entegrasyonu sonrası çözüleceği ve birçok blok zinciri uygulamasının yapay zekâyı güvenilir ve değiştirilemez verilerle destekleyeceğini düşünmekteyiz.

5.1. Kripto Paralar

Kripto paralar blok zinciri teknolojisinin ilk uygulaması olduğu gibi en bilinen uygulamalarıdır. Şu an en bilinen kripto borsalardan biri olan Binance'de yaklaşık 400 coin ve token işlem görmektedir. Günümüzde binlerce coin ve token varlığından bahsedilmektedir. Bu kripto varlıkların tamamından bahsetmemiz mümkün olmadığı için içlerinden bazı önemli kripto paralar burada anlatılacaktır. Bu kripto paraların daha iyi

anlaşılabilmesi için de tarihi gelişimlere göre yaptıkları yenilikler bağlamında konu incelenecektir.

Öncelikle 2009 yılında ilk kripto para olan Bitcoin ortaya çıkmıştır. Ekonomik krizin olduğu bir ortamda üçüncü taraflara ihtiyaç olmadan para transfer sistemi olma iddiasını ortaya koymuştur [9]. Bu proje ile merkeziyetsiz ve güvenli bir sistem kurgulanmıştır. Blok zinciri teknolojisi bu çalışma ile hayatımıza girmiştir. Kripto paralar içerisinde gerçek anlamda en merkeziyetsiz yapının Bitcoin'e ait olduğu söylenebilir. Bölüm 4.1 de ayrıntılı bir şekilde incelendiği için burada çok kapsamlı olarak konuya girilmeyecektir.

2011 yılında Vincent Durham tarafından Bitcoin kaynak kodu kullanılarak yaratılan Namecoin (NMC) Bitcoin'den farklı olarak bir ödeme sistemi değil blok zinciri teknolojisi kullanan bir internet alan adı kayıt sistemi olarak ortaya çıkmıştır. Birçok kaynakta Bitcoin'in ilk çatallı (fork) olarak adlandırılrsa da burada kast edilen blok zinciri çatallanması değildir. Burada aynı koddan türetildiği için bir kod çatallanmasından bahsedilmektedir. Bitcoin ile bağlantısı olmayan tamamen ayrı bir blok zincir olmasına karşın 2013 yılında birleşik madencilik (merged mining) desteği getirilerek Bitcoin madencilerinin Namecoin madenciliği yapması sağlandı. Proje çok yenilikçi ve özgürlükçü bir bakış açısıyla “.bit” uzantılı internet alan adlarının hiçbir kuruluşa bağlı olmaksızın alınabilmesini amaçlamıştır. (”.com” ve “.org” gibi alan adları ICANN (Internet Corporation for Assigned Names and Numbers) gibi merkezi organizasyonlar tarafından yönetilmektedir.) Ancak proje beklenen ilgiyi görmemiştir. Hala aktif olmasına karşın unutulmuş bir proje olarak hayatına devam etmektedir. Namecoin'in bir whitepaper'ı yoktur. Web sayfasında whitepaper kısmında bazı teknik dokümanlar ve form sitelerindeki tartışmalara yer verilmektedir [64]

Yine 2011 yılında Charlie Lee tarafından Bitcoin'in düşük işlem hızı ve yüksek işlem ücretlerine alternatif olarak Litecoin (LTC) tanıtılmıştır. Çok büyük farklılıklar sunmamasına karşın blok süreleri 2,5 dk ya düşürülmüştür. Madencilikte de SHA-256 yerine Scrypt madencilik algoritması kullanarak CPU ve GPU madenciliğini teşvik edilse de 2014 yılından sonra çıkan Scrypt ASIC cihazları sebebiyle CPU ve GPU madenciliği anlamsız hale gelmiştir. Bitcoin için altın benzetmesi yapılırken Litecoin için gümüş benzetmesi yapılmaktadır. En eski kripto paralardan biri olmasına karşın her zaman en büyük işlem

hacmine sahip kripto paralar listesinde yer almaktadır. Bitcoin tabanlı bir yazılım olduğu için birçok yeni teknolojinin Bitcoin’de denenmeden önce denendiği bir ortam sağlamakta bu da popülaritesini korumasını sağlamaktadır. 2017 yılında Bitcoin’den önce SegWit güncellemesini etkinleştirmiştir. Lightning Network’u destekleyen ilk projelerden biri olmuştur. 2021 yılında Mimblewimble güncellemesi ile gizlilik özelliği kazanmış ancak bu özellik yaygınlaşmamış opsiyonel olarak kalmıştır. İnternet sayfasında oldukça bilgi paylaşımı yapılmakta ve önemli kripto paralar arasında olma iddiasını sürdürdüğü görülmektedir [65].

2012 yılında Ryan Fugger, Chris Larsen ve Jed McCaleb tarafından kurulan ve geleneksel kripto paralardan farklılaşan bir proje olan Ripple tanıtılmıştır. Kendisi bir ödeme aracı olmak değil bir ödeme sistemi geliştirmek iddiasındadır. Burada Ripple şirketin adı XRP ise kripto paranın adı olmaktadır. Sistemin temelleri 2004 yılında “RipplePay” olarak Ryan Fugger tarafından atılsada proje 2012 yılında gelişmiş bir sisteme dönüştürülmüştür. Geleneksel blok zinciri yerine XRP Ledger (XRPL) sistemi kullanılmaktadır. PoW ya da PoS gibi mekanizmalar yerine yetkilendirilmiş konsensüs (Federated Consensus) mekanizması ile çalışılmaktadır. Madencilik yoktur çünkü 100 milyar XRP önceden oluşturulmuştur. Aslında bir blok zinciri değildir ama dağıtık defter teknolojisi kullanılmaktadır. Ripple finans sektörü için özel olarak tasarlanmış bir projedir. İşlem hızı yüksek ve işlem maliyetleri çok düşüktür. SWIFT gibi geleneksel sistemlere kıyasla ucuz ve hızlıdır. Ancak merkeziyetsizlik konusunda oldukça eleştiri almaktadır. Bütün coinleri önceden üretilmiştir ve büyük çoğunluğu Ripple Labs’in kontrolündedir. Bu bakış açısıyla merkezi bir finans sistemine benzediği düşünülmektedir. Ripple için özel olarak düzenlenmiş bir whitepaper’ı yoktur ancak açık kaynak kodludur ve ayrıntılı teknik bilgi paylaşılan bir internet sayfası vardır [66]. Ayrıca Ripple kullanmak isteyenlere her türlü bilginin ayrıntılı olarak anlatıldığı resmi bir internet sayfası mevcuttur [67].

2012 yılında Bitcoin bilinirliğini artırmaya başlamıştır. İnsanların Bitcoin’e olan ilgisi madencilik yapmayı yaygınlaştırmış ve bu durum giderek artan bir elektrik tüketimine sebep olmaya başlamıştır. Bu konuda yeni bir çözümü Peercoin getirmiştir. Konsensüs mekanizması olarak PoW yerine elektrik tüketimini artırmayacak bir çözüm sunulmuştur.

Peercoin (PPC) 2012 yılında Sunny King ve Scott Nadal tarafından Bitcoin açık kaynak kodundan türetilmiş bir kripto paradır. PoW ile birlikte PoS kullanan ilk coin olması sebebiyle oldukça önemli bir projedir. Coin üretimi sınırsız olmasına karşın PoS teknolojisi ile yıllık enflasyonu yaklaşık %1 olarak hesaplanmaktadır. Bu konulara “PPcoin:Peer-to-Peer Crypto- Currency with Proof-of-Stake” isimli whitepaper’ında ayrıntılı olarak yer verilmiştir [68]. Coin sahiplerinin tuttukları bu coinler sayesinde madencilik yapması fikri zamanla birçok farklı coinde kullanılmaya başlanmıştır. Peercoin günümüzde hala aktif olmasına karşın popülerliği oldukça azalmış bir kripto paradır. Whitepaper ve ilgili diğer dokümanlarına internet sitesinden ulaşılabilir [69].

Dogecoin (DOGE) kripto para dünyasının anlaşılması açısından çok güzel bir örnektir. 2013 yılında Billy Markus ve Jackson Palmer tarafından kurulmuştur. O dönemde kripto paralara yoğunlaşan ilgiyi mizahi olarak ortaya koymak için şaka amaçlı yaratılmış bir kripto paradır. Litecoin’in bir kopyasıdır ve teknik açıdan hiçbir yenilik sunmamıştır. Ancak kısa zamanda çok fazla ilgi görerek hiç beklenmedik bir çıkış yapmıştır. Kripto paralar için önemi ise aslında insanların kabul ettiği şeylerin arkalarında bir teknoloji ya da yenilik barındırmasa bile değerli olabileceği gerçeğidir. Günümüzde değer atfedilen birçok varlık için bu söylenebilir. O yüzden kripto paraların yeteri kadar insan tarafından kabul görmesi halinde diğer paraların yerini alabileceği iddiasını doğrulayan bir proje olarak değerlendirilebilir. Günümüzde hala oldukça büyük bir piyasa değerine sahiptir. Resmi sitesinde Dogecoin’in insanları gülümseten tesadüfi bir kripto hareketi olduğu ve resmi olmayan sloganlarının “Do Only Good Everyday” herşeyi özetlediği belirtilmektedir [70]

Bitcoin’in şeffaflık ve izlenebilirlik konusundaki duruşu herkes tarafından benimsenmemiştir. Yapılan işlemlerin tamamen gizli ve takip edilemez olması gerektiğini savunan birçok kişi olmuş bu konu üzerine protokoller geliştirilmiştir. Ancak 2013 yılında Nicolas van Saberhagen takma adıyla yayımlanan “CryptoNote v 2.0” makalesi bu konuda en başarılı projeyi gerçekleştirmiş olan Monero (XMR) projesinin whitepaper’ı olarak görülmektedir [71]. Bu makale sonrası Bytecoin (BCN) piyasaya sürülmüş, daha sonra çatallanma ile “BitMonero” adlı yeni bir kripto para oluşturulmuş bir süre sonra da bu paranın ismi Monero’ya dönüştürülmüştür. Bu proje kendisinden önceki projelerden farklı olarak tüm işlemleri gizli tutmaktadır. Halka imzaları (Ring Signatures) yöntemiyle işlemi

gerçekleştiren kullanıcı tespit edilemez olmaktadır. Gizli Adresler (Stealth Addresses) yöntemiyle alıcı adresleri görüntülenememekte ve alıcının kimliğinin tespiti ve işlemin takip edilmesi önlenmektedir. İşlemlerde kaç XMR kullanıldığı sadece taraflar tarafından görülebilmektedir. Sonuç olarak Monero işlemleri tamamen gizli ve anonimdir. PoW konsensüs mekanizmasını kullanmakla beraber SHA-256 yerine ASIC cihazlara karşı dirençli Randomx algoritması geliştirilmiştir. Bu durum CPU ve GPU madenciliğini ön planda tuttuğu için madenciliğin merkezi hale gelmesini ve madencilik havuzlarının sistemi kontrol etmesini önlemektedir. Monero hala aktif olarak kullanılan önemli bir proje olmakla birlikte kara para aklama ve yasa dışı işlemlerde kullanılmaya çok müsait olduğu için resmi otoritelerce eleştirilen bir projedir. Resmi sitesinden oldukça ayrıntılı bilgilere erişilebilmektedir [45].

Dash Coin (DASH), 2014 yılında Evan Duffield tarafından, Bitcoin'den farklı olarak düşük maliyet, yüksek işlem hızı, gizlilik odaklı ödemeler, anonim işlemler ve merkeziyetsiz yönetim sunmayı amaçlayan bir kripto para olarak geliştirilmiştir. Başlangıçta Darkcoin olarak adlandırılan Dash, 2015 yılında yeniden markalaşarak "Digital Cash" (Dijital Nakit) anlamına gelen Dash adını almıştır. Masternode Sistemi (İki katmanlı Ağ) ile çalışmaktadır. Masternode'lar 1000 DASH stake eden kullanıcılardan oluşur. Bu kurgu sayesinde ağ daha hızlı ve merkeziyetsiz çalışır. Instant Send (Anında Gönderim) özelliği sayesinde Masternode'lar aracılığı ile işlemler blok onayı beklenmeden anında onaylanabilir. Private Send (Gizli Gönderim) özelliği ile işlemler takip edilemez hale getirilebilmektedir. Merkeziyetsiz otonom Organizasyon (DAO) modelini uygulayan ilk projelerden biridir. Alınan kararlar topluluk ve Masternode sahipleri tarafından belirlenmektedir. İşlem ücretleri çok düşük seviyelerdedir. PoW algoritması kullanılmaktadır ancak blok ödülllerinin %45'ni madenciler alırken %45'ni Masternode'lar alır. Geriye kalan %10 ekosistemin geliştirilmesi için DAO fonuna gider [72].

Bitcoin'in yayımlanması sonrası birçok insan Bitcoin'i anlamaya ve bu teknolojiyi irdelemeye başlamıştır. Bunlardan birisi de Vitalik Buterin dir. Kendisi sıkı bir Bitcoin savunucusu olarak birçok makale yazmış ve Bitcoin hakkında en eski ve saygın yayınlardan olan "Bitcoin Magazine"i kurmuştur. (Mihai Alisie ile birlikte). Bu süreçte bazı konularda Bitcoin'in yetersiz kaldığını görmüştür. Bitcoin'in sadece işlemleri kaydetmesinin, otomatik

bir kod çalıştırmamasının önemli bir eksiklik olduğunu, zira bu teknoloji ile çok karmaşık işlemlerin otomatikleştirilebileceğini düşünmüştür. Bitcoin esnek değildir. Değişikliklere ve yeni önerilere hep mesafeli ve hantal kalmaktadır. Bunun sebebinin teknolojisinin kendi kısıtları ve Bitcoin topluluğunun oldukça muhafazakâr davranması olduğunu değerlendirmiştir. Vitalik gördüğü bu eksikliklere karşı kendi yeni bir teknoloji sunmuş ve 2013 yılında Ethereum whitepaper'ını "Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform" [59] yayımlamıştır. Bu makale incelendiğinde Vitalik'in bütün eleştirileri ve ayrıntılı çözümleri makalede görülmektedir. Ethereum oldukça devrimsel bir projedir. Blok zinciri teknolojisiyle kripto para dışında daha birçok farklı şeyin yapılabileceğini ortaya koymuştur. Ethereum ile; akıllı sözleşmeler, merkeziyetsiz kuruluşlar, merkeziyetsiz veri depolama, merkeziyetsiz hesaplama gibi onlarca farklı projenin yaratılabileceği bir platform sunmuştur. Ethereum 2015 yılında faaliyete geçmiştir. Ethereum ve akıllı kontraktlar hakkında detaylı bilgiler Bölüm 4.2 de sunulmuştur.

2016 yılında Zooko Wilcox-O'Hearn tarafından geliştirilen Zcash, Bitcoin'in açık ve izlenebilir işlem modeline alternatif olarak, kullanıcıların işlemlerini gizleyebilmesine olanak tanıyan bir protokol sunmuştur. Zcash i farklı kılan gizlilik çalışmalarından ziyade Zero Knowledge Proofs (ZKP- Sıfır Bilgi İspatı) ile zk-SNARKs (Zero-Knowledge Succinct Non-Interactive Argument of Knowledge) teknolojisini kullanan ilk büyük kripto para projesi olmasıdır. Bu teknolojide bir işlemin doğruluğunun kanıtlanması için karşı tarafın ayrıntıları görmesine gerek yoktur. Gönderen ve alıcı adresleri ve işlem tutarı gizlenebilir. Bu kapsamda bu projenin whitepaper'ı olan "Zerocash: Decentralized Anonymous Payments from Bitcoin" adlı makale oldukça kapsamlı matematiksel açıklamalar içeren uzun bir akademik çalışmadır. [48] Proje Bitcoin kod tabanından üretilmiştir. PoW konsensüs algoritması kullanmaktadır. Kullanıcılar şeffaf işlemler (T-Tx) ya da korunan işlemler (Z-Tx) tercih edebilirler. Madencilik Bitcoin gibi olmakla birlikte SHA-256 yerine Equihash algoritması kullanmaktadır. 2020 yılından madencilik ödüllerinin %20'sini geliştirme fonuna ayrılmaya başlanmıştır. Daha ayrıntılı bilgilere projenin resmi sitesinden ulaşılabilir [73].

Blok zinciri teknolojisinin benimsenerek yeni ürünler üretme çalışmalarının yoğunlaştığı dönemin en güzel örneklerinden biri de STEEM projesidir. Web 3.0 yaklaşımına en uygun çalışma olarak değerlendirebileceğimiz bu proje merkeziyetsiz bir sosyal medya ve içerik platformu olarak 2016 yılında Dan Larimer ve Ned Scott tarafından geliştirilmiş. Kullanıcıların içerik üreterek, paylaşarak ve beğenerek ödüller kazanmalarını sağlayan ilk blok zinciri tabanlı sosyal medya ekosistemidir. Projenin hayata geçmesinden bir yılı geçkin bir süre sonra 2017 yılında “Steem: An incentivized, blockchain-based, public content platform” [74] isimli oldukça ayrıntılı bilgi içeren whitepaper’ı yayımlanmıştır. Ancak ilginç bir şekilde bu çalışmada yazarların isimleri bulunmamaktadır. Sansüre karşı dayanıklı, kullanıcıların sadece paylaşım yaparak kripto kazanabildiği ve içerikler ile kullanıcıların topluluk tarafından oylarla değerlendirildiği bir projedir. DPoS konsensüs algoritmasını kullanmaktadır. Her 3 saniyede bir blok üretebilmektedir. Üç katmanlı bir token sistemi ile çalışmaktadır. Steem temel kripto para sistemidir ve borsalarda alınıp satılabilir. Steem Power (SP) Steem’in stake edilmiş halidir ve ne kadar SP’ye sahip olunursa içerik oylama gücü o kadar fazla olmaktadır. Steem Dollars (SBD) 1 ABD dolarına eşit olmasına çalışılan stabilcoin mantığında bir tokendir. Steemit platformundaki içerik ödülleri SBD olarak verilmektedir. SBD Steem ile takas edilebilir ya da borsalarda satılabilir. Bu ekosistemde içerik paylaşarak, içerik tüketerek ve oylayarak token kazanılmaktadır. Projenin resmi internet sayfasından daha ayrıntılı bilgilere ulaşılabilmektedir [75].

Yine 2017 yılında Ethereum’a alternatif olan ve ilk akademik kripto para olarak değerlendirilen Cardano projesi hayata geçirilmiştir. Proje bir whitepaper yayımlamamış, bunun yerine protokolü oluşturan her bileşen için hakemli makaleler ve akademik çalışmalar sunulmuştur. Ethereum’un kurucularından Charles Hoskinson tarafından geliştirilen bu proje, akademik olarak inşa edilen bir mimariye sahiptir. Cardano iki katmanlı çalışan bir yapıya sahiptir: Cardano Settlement Layer (CSL) ve Cardano Computational Layer (CCL). CSL katmanında, projenin yerel kripto para birimi olan ADA işlemleri gerçekleştirilmektedir. ADA, hem ağ üzerindeki ödemelerde hem de stake işlemlerinde kullanılan, PoS temelli güvenlik modelinde doğrulayıcıları teşvik eden temel ekonomik varlık konumundadır. CCL katmanı ise akıllı sözleşmelerin çalıştığı bölümdür. Cardano, konsensüs algoritması olarak, matematiksel olarak kanıtlanmış ilk PoS protokolü olan Ouroboros’u kullanmaktadır. Akıllı sözleşme yazımı için ise Plutus ve Marlowe adlı kendi programlama dillerini geliştirmiştir. Bu katmanlı yapı ve özgün protokoller sayesinde daha

esnek, güvenli ve hızlı bir blok zinciri mimarisi hedeflenmiştir. Proje hakkında daha ayrıntılı teknik bilgilere resmi internet sitesinden ulaşmak mümkündür [76].

Kripto borsaların büyümesi ile birlikte bazı borsalar kendi ekosistemlerinde kullanmak üzere kripto paralar oluşturdular. Bunlardan biri de 2017 yılında Ethereum üzerinde ERC-20 token olarak Changpeng Zhao tarafından çıkarılan Binance Coin (BNP) dir. Binance borsasında işlem ücretlerinde indirim sağlanması için oluşturulmuş bir proje olmakla birlikte, Binance'nin büyümesi ve daha güçlü hale gelmesi ile 2019 yılında Ethereum dan ayrılarak kendi blok zinciri olan Binance Chain'e taşınmıştır. Binance Chain yüksek hızlı ve düşük maliyetli merkeziyetsiz çalışan bir borsa (DEX) oluşturmak için geliştirilmiştir. Daha sonrasında ise Binance Smart Chain devreye sokularak Ethereum'a alternatif akıllı sözleşmeleri destekleyen bir ağ oluşturulmuştur. Böylelikle BNB geniş bir blok zinciri ekosistemine dönüşmüştür. Binance'nin en büyük kripto borsalardan biri haline gelmesi uygun fiyat ve sunduğu hizmetlerdeki çeşitlilik sayesinde ve böylece oldukça güçlü bir ekosistem haline gelmiştir. BNB yakım mekanizması ile deflasyonist bir yaklaşım benimsemiştir. Bu bilgilere ve daha fazlasına resmi sitesinden ulaşılabilir [77].

2017 yılının kripto dünyasında en önemli gelişmelerinden biri de akıllı sözleşmelerin gerçek dünya verilerine erişebilmesini sağlayacak merkeziyetsiz bir oracle ağının kurulmasıdır. Ayrıntıları "ChainLink: A decentralized Oracle Network" [78] adlı whitepaper ile duyurulan bu proje Sergey Nazarov ve Steve Ellis tarafından kurulmuştur. Geleneksel merkezi oracle çözümleri yerine merkeziyetsiz bir oracle çözümü sunulmaktadır. DeFi, sigorta, oyun ve diğer sektörlerde akıllı sözleşmelerin dış dünyadan güvenilir veri alması sağlanmaktadır. Chainlink bu verileri otomatik olarak sağlamaktadır. Kripto para fiyatları, döviz kurları, emtia fiyatları, hava durumu verileri, spor karşılaşma sonuçları gibi hizmetleri resmi sitelerin apilerinden alarak akıllı sözleşmelere sunmaktadır. Yani istenilen her veriye ulaşılması mümkün olmamakta sadece Chainlink tarafından sunulan veriler kullanılabilir. Bir sözleşme bilgi talebinde bulununca bu talep Chainlink ağı tarafından bağımsız oracle node'larına iletilir. Oracle nodeları dış dünyadan veri toplar. Toplanan bu verilere doğruluk kontrolü yapılarak talep yapan sözleşmeye gönderilir. Chainlink başlangıçta Ethereum ağına bir ERC-20 token olarak oluşturulmuştur. Sonrasında Solana, Polygon, Binance Smart Chain (BSC), Avalanche gibi diğer birçok blok zincirde de kullanılabilir hale

gelmiştir. Kripto parası Link'tir. Farklı zincirlerde üretilen Link'ler aynı olmamakla birlikte birbirlerine çevrilebilmektedirler. Her zincir üzerinde ayrı düğümler çalışmakla birlikte bütün oracle düğümleri Chainlink ağında çalışmaktadır. Daha fazla ayrıntı bilgiye projenin internet sayfasından ulaşılabilir [79].

Ethereum'un kurucularından biri olan Dr. Gavin Wood tarafından 2016 yılında whitepaper'ı "Polkadot: Vision for a Heterogeneous Multi-Chain Framework" [80] yayımlanan Polkadot ağı 2020 yılında faaliyete başlamıştır. Sunduğu yenilik farklı blok zincirler için birlikte çalışabilirlik (interoperability) sağlayıp güvenli ve ölçeklenebilir bir ekosistem kurmaktır. Ana Zincir (Relay Chain) çekirdek zincirdir ve tüm Parachain'lerin güvenliğini, işlemlerin koordinesini ve Parachain'ler arasında veri transferini sağlamaktadır. Ana Zincir'de akıllı kontratlar desteklenmez Parachain'lerde çalıştırılabilirler. Parachain'ler kendi özelleştirilmiş kurallarına ve özelliklerine sahip bağımsız blok zincirleridir. Merkeziyetsiz uygulamaların (DApp'ler) ve özel blok zincirlerin Polkadot ağı üzerinde çalışmasını sağlarlar. Farklı Parachainler aynı anda işlem yapabilirler bu Polkadot ağının çok verimli çalıştığını göstermektedir. Parachain'ler açık artırma (auction) yönetimi ile kiralanırlar. Parathread'ler küçük projeler için Parachain'lerin daha ekonomik bir versiyonu olarak kullanılır ancak bu yapıda kullanıldığı kadar ödeme yapılmaktadır. Polkadot ağının farklı blok zincirleri bağlamak için kendi köprüleri (Bridges) vardır. Ağ hard fork yapmadan güncellenebilmektedir. Ağ güncellemeleri topluluk onayıyla (on-chain governance) gerçekleştirilmektedir. Ağın kripto para birimi olan DOT tokenini stake eden kullanıcılar ağın güvenliğini sağlar ve ödül kazanırlar. NPoS (Nominated Proof-of-Stake) adlı özgün bir konsensüs mekanizması kullanılmaktadır. Ayrıntılı bilgilere projenin resmi internet sayfasından ulaşılabilir [81].

Ölçeklenebilmenin ciddi bir problem olması Ethereum gibi popüler ağlarda Katman 2 çözümlerinin (Layer 2) gelişmesini sağlamıştır. Bu sorunları Katman 1 de çözmeye çalışan yeni projeler ortaya çıkmıştır. Bu projelerin en önemlilerinden bir tanesi saniyede 65.000 işlem yapma kapasitesine sahip Solana'dır. 2017 yılında Anatoly Yakovenko tarafından geliştirilmeye başlanmış, 2020 yılında proje aktif hale gelmiştir. 2017 yılından "Solana: A new architecture for a high performance blockchain" [82] adlı whitepaper yayımlanmış ve Solana'nın bu hızlara ulaşmasını sağlayan temel faktör olan PoH (Proof of History)

konsensüs mekanizması bu makalede tanıtılmıştır. PoH’da işlemler kriptografik gecikme fonksiyonu (Verifiable Delay Function – VDF) ile sıralanır. Her işlem önceki işlemin hash’ine dayanır. Böylece işlemlerin değiştirilemez bir sırası oluşturulur. Zaman damgalama (Timestamping) ile ağ işlemleri belirlenmiş bir sıraya göre işlemektedir. Doğrulayıcılar (Validatörler) zaman damgaları sayesinde sıralı aldıkları işlemleri çok hızlı doğrularlar. Kullanıcıların doğrulayıcı olması için sistemin kripto parası “SOL” stake etmeleri gerekir. Ağ güvenliği PoH ile birlikte kullanılan PoS konsensüs mekanizması ile sağlanır. Rust ve C programlama dillerinde yazılan akıllı sözleşmeleri destekler. Oldukça hızlı işlem süreleri ve düşük işlem ücreti sunabilmektedir. Tarihinde yaşadığı ağ kesintileri sebebiyle oldukça eleştirilmektedir. Solana’nın ekosisteminde birçok farklı proje yer almaktadır. “Serum” ve “Raydium” merkeziyetsiz borsaları, “Magic Eden” NFT pazarı, “Phantom” kripto cüzdanı solana ekosisteminde çalışmaktadır. Daha ayrıntılı bilgilere projenin resmi internet sayfasından ulaşılabilir [83].

Blok zinciri teknolojisinin gelişmesiyle birlikte zincirlerin boyutu ve ağın her katılımcı tarafından tam olarak saklanması gibi problemler ortaya çıkmıştır. Bu probleme en yenilikçi çözümlerden birini getirme iddiasında olan proje Mina Protocol’dür. Mina, 2017 yılında Evan Shapiro ve Izaak Meckler öncülüğünde geliştirilmeye başlanmış ve 2021 yılında ana ağı faaliyete geçirilmiştir. Mina’nın temel amacı, blok zincirinin boyutunu birkaç kilobayt seviyesinde sabit tutarak herkesin tam düğüm çalıştırabildiği, ultra hafif bir blok zinciri oluşturmaktır. Bunu mümkün kılmak için zk-SNARK teknolojisini ana katmana entegre etmiş ve her yeni blokta, önceki blokların tüm geçmişini kriptografik olarak ispatlayan bir “zero-knowledge proof” oluşturulmuştur. Böylece ağdaki düğümlerin zincirin tamamını indirmesine gerek kalmamaktadır. Mina bu yönüyle blok zinciri değil, zincirin kriptografik “kanıtı” üzerinden çalışan yeni bir yaklaşım sunmaktadır. PoS (Proof of Stake) konsensüs algoritması ile çalışan Mina ağı, işlem doğrulaması yapan kullanıcıları “block producer” olarak tanımlamaktadır. Mina aynı zamanda zkApps adı verilen, özel verilerle etkileşim kurabilen sıfır bilgi ispatlarına dayalı akıllı sözleşmelere de olanak tanımaktadır. Bu yönüyle gizliliği ön planda tutarak kullanıcı verilerini ifşa etmeden işlem yapılabilmesini sağlanmaktadır. Projenin whitepaper’ı olan “The Mina Protocol: Lightweight Blockchain with Recursive zk-SNARKs” başlıklı belge oldukça teknik detaylar içermektedir. Projenin resmi internet sitesinde, whitepaper ve bu teknolojilerin uygulama örnekleriyle anlatıldığı ayrıntılı dökümantasyonlar da yer almaktadır [84].

IOTA, blok zinciri tabanlı olmayan, ancak dağıtık defter teknolojisi (DLT) üzerine kurulu alternatif bir mimari olarak değerlendirilmektedir. 2015 yılında David Sonstebo, Sergey Ivancheglo, Dominik Schiener ve Serguei Popov tarafından kurulan proje, özellikle Nesnelerin İnterneti (IoT) uygulamaları için hızlı, ölçeklenebilir ve işlem ücreti olmayan bir dijital ödeme sistemi sunmayı hedeflemiştir. IOTA'nın altyapısında kullanılan “Tangle” isimli yapı, klasik blok zinciri yerine yönlendirilmiş çevrimsiz grafik (DAG) mimarisine dayanmaktadır. Bu sistemde her yeni işlem, önceki iki işlemi doğrulamak zorundadır; böylece madencilere gerek kalmadan ağ kendi kendini sürdürebilir hale gelmektedir. IOTA’da işlem ücretinin olmaması, mikro ödemeler için ideal bir ortam yaratmaktadır. Proje, merkeziyetsizlik hedefiyle yola çıkmış olsa da uzun süre boyunca ağın güvenliği “koordinatör” adlı merkezi bir yapı ile sağlanmıştır. Bu durum, özellikle akademik çevrelerde eleştiri konusu olmuş ve “Coordicide” adlı güncellemeyle merkezi yapının ortadan kaldırılması planlanmıştır. IOTA, PoW ya da PoS konsensüs algoritmalarını kullanmamaktadır. Bunun yerine doğrulama ve işlem üretimi kullanıcılar arasında paylaşılmaktadır. Bu sistem mimarisi sayesinde IOTA, enerji tüketimi açısından oldukça verimli bir yapı sunmaktadır. Whitepaper’ı olan “The Tangle” başlıklı teknik döküman, matematiksel modelleme açısından zengin bir içeriğe sahiptir. Ayrıca projenin resmi internet sitesinde IOTA Foundation tarafından hazırlanan geliştirme kılavuzları ve dokümantasyonlar da kullanıcıların erişimine sunulmuştur [85].

İncelenmesi gereken bir başka proje de yüksek işlem hızı, düşük işlem maliyeti ve özelleştirilebilir alt ağ (subnet) mimarisi ile öne çıkan Avalanche ağıdır. Projenin platform yapısını, konsensüs algoritmasını, token ekonomisini ve stablecoin dizaynını ayrıntılarıyla açıklayan dört ayrı whitepaper yayımlanmıştır [86]. Tüm bu geliştirmeler, merkezi ABD’de bulunan ancak kurucusu Türk bilgisayar mühendisi Prof. Dr. Emin Gür Sirer olan Ava Labs şirketi tarafından gerçekleştirilmiştir. Avalanche, ölçeklenebilirlik ve uygulama çeşitliliği açısından oldukça yenilikçi bir mimari sunmaktadır.

Ağın en dikkat çekici özelliği, üçlü blok zinciri yapısıdır:

- X-Chain (Exchange Chain): Dijital varlıkların oluşturulması ve transferinden sorumludur.

- C-Chain (Contract Chain): Ethereum uyumlu akıllı sözleşmelerin çalıştırıldığı zincirdir. Ethereum geliştirici araçları (Solidity, MetaMask, Remix) doğrudan desteklenir.
- P-Chain (Platform Chain): Alt ağların (subnet) ve doğrulayıcıların yönetildiği, staking işlemlerinin koordine edildiği zincirdir.

Avalanche ağı, kendi adını taşıyan Avalanche Konsensüs Mekanizması ile çalışmaktadır. Bu mekanizma, klasik BFT protokollerinden farklı olarak olasılıksal karar alma süreciyle, ağdaki işlemleri saniyenin altında kesinliğe ulaştırabilmektedir. Bu mimari sayesinde hem yüksek güvenlik hem de yüksek işlem hacmi bir arada sağlanmaktadır. Projenin yerel kripto para birimi olan AVAX, sistemin ekonomik yapısının merkezinde yer almaktadır. AVAX; işlem ücretlerinin ödenmesinde, staking yoluyla doğrulayıcı olarak ağa katılmada ve alt ağlar oluşturulurken güvenlik teminatı olarak kullanılmaktadır. Ayrıca sistemde, her işlemde belirli miktarda AVAX yakılarak (burn) toplam arz zaman içinde azaltılır; bu yönüyle deflasyonist bir model benimsenmiştir.

Avalanche'ın bu teknik yapısı, 7. Bölümde bahsedilen tarafımızca geliştirilen tokenize edilmiş araç kiralama modeli gibi işlemlerin, düşük maliyetle ve yüksek hızla gerçekleştirilmesi gereken senaryolar için son derece uygundur. Akıllı sözleşme katmanı olan C-Chain'in Ethereum ile uyumlu olması, Solidity diliyle yazılmış sözleşmelerin doğrudan entegre edilmesini sağlamaktadır. Ayrıca, uygulamamızda planlanan farklı araç gruplarının kendi alt ağları (subnet) içinde izole çalışabilmesi, regülasyonlara uyum ve kurumsal ayrıştırma açısından büyük avantaj sağlamaktadır. Alt ağlar aracılığıyla, örneğin sadece Avrupa'da faaliyet gösterecek kiralama şirketleri için Avrupa veri mevzuatlarına özel bir subnet kurulması, veya daha düşük işlem maliyetine sahip subnet'ler üzerinde düşük bütçeli araçların listelenmesi mümkündür. Bu da Avalanche'ın sadece teknik değil aynı zamanda hukuki ve ekonomik uyarlanabilirlik açısından da esnek bir yapı sunduğunu göstermektedir. Avalanche projesinin tüm teknik detaylarına, geliştirici kılavuzlarına ve whitepaper belgelerine Avalanche'ın resmi internet sitesinden ulaşılabilir [87].

5.2. Token ve NFT

Token akıllı sözleşme destekleyen bir blok zinciri üzerinde bir varlığı, değeri ya da hakkı temsil eden bir kripto varlıktır. Akıllı sözleşme desteklemeyen blok zincirlerde ek katmanlarla da token yaratılabilmektedir. Token kavramı üç temel başlık altında incelenmektedir. Değiştirilebilir (Fungible) Tokenler, Değiştirilemez (Non-Fungible) Tokenler yani NFT'ler ve Hibrit (Hybrid) Token'lar. Değiştirilebilir tokenler birbirinin yerine geçebilen ve para gibi çalışabilen tokenlerdir. Yani 1 Link başka 1 Link'e eşittir. Her zincirin bu tip token üretmek için kendi standardı vardır. (ERC-20 Ethereum, BEP-20 BSC, TRC-20 Tron, SPL Solana) Fungible Tokenleri 5 başlık altında inceleyebiliriz;

Ödeme Token'ları (Payment Tokens): Ana amaçları ödeme ve değer transferidir. Bazı kaynaklar coinleri de fonksiyonel açıdan bu başlık altında gösterse de kanaatimizce coinler token olarak değerlendirilmemelidir. Ödeme tokenlerine USDT, USDC, DAI, BUSD örnek olarak verilebilir.

Fayda Token'ları (Utility Tokens): Bir platform veya hizmete erişim sağlamak için kullanılmaktadır. Doğrudan bir yatırım ya da değer transferi için kullanılmamaktadır. Ethereum ağında Basic Attention Token (BAT) ve Solana ağında Audius (AUDIO) örnek olarak verilebilir.

Menkul Kıymet Token'ları (Security Tokens): Hisse senedi, tahvil, bono ve emtialar gibi menkul kıymet enstrümanlarının blok zincir üzerindeki karşılığı olarak değerlendirilebilir.

Ethereum ağında (ERC-1404) ABD SEC onaylı halka arz ile çıkarılan INX Token ve Ethereum ağında Overstock şirketine ait tZERO platformunda, temettü hakkı içeren tZERO (TZROP) örnek olarak verilebilir.

Stabil Kripto Paralar (Stablecoins): Değeri sabit varlıklara endekslenmiş token'lardır. 1:1 oranında fiat para birimine sahiplerdir. Örneğin 1 USDT = 1 USD

Varlığa Dayalı Token'lar (Asset-Backed Tokens): gerçek dünyadaki varlıkların tokenize etmek için kullanılmaktadır. Altın, gayrimenkul, petrol gibi fiziksel varlıkları blok zincirde temsil eder. Örneğin Paxos Gold (PAXG) için 1 PAXG = 1 ons altındır.

Non-Fungible Token'ler (NFT - Değiştirilemez Token'ler) ise her biri benzersiz olan ve değiştirilemez olan tokenlerdir. TDK NFT için Türkçe karşılık olarak “Nitelikli Fikri Tapu” terimini belirlemiştir. Sanat, oyun, metaverse, kimlik doğrulama gibi alanlarda kullanılabilirler. Blok zincirlerinin bu tokenler için de teknik standartları oluşturulmuştur (ERC-721, ERC-1155 Ethereum, BEP-721 (BSC), SPL NFT Solana).

NFT ler bir dönem çok popüler olmuş 2022 yılında işlem hacmi olarak tarihi zirveye ulaşmış ve daha sonra popüleritesini kaybetmiştir. NFT çılgınlığı denilebilecek bu dönemi dolandırıcılık olarak da tanımlayan birçok kişi olmuştur.

Hybrid Token (Hibrit Token) ise hem fungible hem de non-fungible özelliklere sahip token'lardır (Ethereum ERC-1155). Örnek olarak, bir oyunda fungible token'lar oyun içi para olarak kullanılabilirken, non-fungible token'lar (NFT'ler) özel karakterler, silahlar veya araziler olarak temsil edilebilir. Bir başka örnek olarak, bir bina tokenize edildiğinde, her daire NFT olarak saklanırken, tüm bina için hisse sahipliği fungible token olarak dağıtılabilir. Ya da tokenize edilmiş varlıkların yönetim hakları farklılaştırılabilir.

5.3. Köprü Yapıları (Bridges)

Köprüler, farklı blok zinciri ağları arasında veri, varlık ve akıllı sözleşme etkileşimini sağlamak için kullanılan protokollerdir. Bu protokoller sayesinde farklı zincirlerden token transferi yapılabilir, farklı zincirlerde likidite havuzları oluşturulabilir, akıllı sözleşmelerin farklı blok zincir akıllı sözleşmeleri ile etkileşimi sağlanabilir ve NFT'ler farklı zincirlere taşınabilir. Köprüler blok zinciri ekosisteminin en önemli bileşenlerinden biri olmakla birlikte en fazla saldırı alan en zayıf noktalarından biridir.

Köprüler temel de iki mantık ile çalışır. Birincisi Kilitleme ve Basma (Lock and Mint) Mekanizması ikincisi ise Likidite Havuzu (Liquidity Pool) Mekanizmasıdır. Birinci mekanizmada bir varlık, kaynağında kilitlenir. Örneğin Wrapped Bitcoin (WBTC) köprüsünü kullanan bir kullanıcı Bitcoin ağındaki BTC'yi bir merkezi saklama cüzdanına kitler. Aynı değerde yeni bir varlık hedef blok zincirinde oluşturulur. Örneğin Aynı değerde WBTC Ethereum ağında yaratılır. Kullanıcı BTC'sini Ethereum ağına taşımış olur. Eğer kullanıcı BTC'sini geri almak isterse oluşturulan WBTC'ler yok edilir ve kilitlenmiş olan BTC serbest bırakılır. Likidite Havuzunda ise iki farklı zincirde oluşturulmuş birbiri ile etkileşimli likidite havuzları bulunmaktadır. Kullanıcının Ethereum ağında yer alan havuza gönderdiği Ether'e eşdeğer Sol, Solana ağında bulunan havuzdan kullanıcıya verilir.

Kilitle ve Basma Mekanizmasında sahip olduğunuz kripto varlığın bire bir eşdeğeri karşı tarafta yaratılır. Farklı blok zincirlerde varlık kullanılabilir ve kullanıcılar her zaman asıl kripto varlıklarına geri dönebilirler. Buna karşılık kullanıcı bir varlık saklayıcısına güvenmek zorundadır ve saklayıcının cüzdanı siber saldırıya uğrarsa yaratılan token'ler değersizleşir.

Likidite havuzları daha hızlı bir çözüm sunar çünkü yeni bir kripto yaratma ve yok etme süreçleri yoktur. Merkeziyetsizdir, çünkü tüm işlemler akıllı sözleşmelerle otomatik olarak yapılır. Kilitlenmiş varlık olmadığı için bu varlıklara saldırı riski taşımaz. Havuzda talep edilen miktarda karşı varlık yoksa işlem gecikir ya da maliyeti artar. Sahip olunan kripto varlıkla alınacak kripto varlık arasındaki parite hesaplamaları her zaman orijinal varlığın tam değerini sağlamayabilir. Havuzların siber saldırıya uğrama ihtimali vardır ve likidite biterse işlem yapılamaz.

5.4. Merkeziyetsiz Finans (DeFi) ve Uygulamaları

Bankalar, aracı kurumlar ve geleneksel finansal yapılar yerine, akıllı sözleşmeler ve blok zinciri teknolojisi kullanılarak finansal hizmetleri blok zinciri ekosistemine taşımayı amaçlayan projelerdir.

Merkeziyetsiz finans, merkezi bir otoritenin bulunmadığı, işlemler blok zinciri üzerinde gerçekleştiği için kullanıcılar tarafından doğrulanabildiği, herhangi bir kişisel bilgi paylaşmadan hizmetlerden yararlanılabildiği ve akıllı sözleşmeler sayesinde üçüncü taraflara gerek kalmadığı bir ekosistem oluşturmaktadır.

Merkeziyetsiz Borsalar (DEX – Decentralized Exchanges) bu alandaki en başarılı uygulamalardan biridir. Akıllı sözleşmelerle çalışan, alıcı ve satıcıların doğrudan etkileşime girdiği ve likidite havuzu kullanan yapılardır. Örnek olarak, Uniswap (Ethereum tabanlı), SushiSwap (Ethereum, BSC, Polygon destekliyor), PancakeSwap (Binance Smart Chain - BSC tabanlı) verilebilir.

Bir başka DeFi örneği Borç Alma ve Borç Verme (Lending and Borrowing) uygulamalarıdır. Bir kredi kuruluşuna ihtiyaç olmadan kullanıcılar birbirlerine borç verebilmektedirler. Kripto varlıklar teminat gösterilerek borç alınabilmektedir. Faiz oranları otomatik olarak akıllı sözleşmelerle belirlenmektedir. Örnek olarak, Aave (Ethereum tabanlı kredi protokolü), Compound (Merkeziyetsiz borç alma/verme platformu), MakerDAO (DAI stablecoin'i oluşturmak için kullanılır) verilebilir.

DeFi ekosisteminde volatilitiyi azaltmak için algoritmik ya da rezerv teminata dayalı çalışan Stabil Kripto Paralar (Stablecoins) ve kullanıcıların varlıklarını DeFi protokollerinde stake ederek pasif gelir elde ettikleri Getiri Çiftçiliği (Yield Farming) uygulamaları da DeFi örnekleridir.

Önemli bir başka DeFi uygulaması da siber saldırılar ve protokol hatalarına alternatif çözüm olarak geliştirilen Merkeziyetsiz Sigortalardır (Decentralized Insurance). Örnek olarak, Nexus Mutual (DeFi protokolleri için sigorta hizmeti sağlar) ve Cover Protocol (DeFi sigorta platformu) verilebilir.

5.5. Merkeziyetsiz Otonom Organizasyonlar (DAO'lar)

DAO'lar merkezi bir yönetici ya da CEO yerine akıllı sözleşmelerle yönetilen, karar alma süreçlerini tamamen topluluğa bırakan ve katılımcıların oy kullanarak yönetim kararlarını aldığı blok zinciri tabanlı organizasyonlardır.

DAO'lar akıllı sözleşmelerle kuruldukları için oldukça farklı yaklaşımlarla kurulmaları mümkündür ancak organizasyonun nasıl çalışacağı, hangi oylama sistemini kullanacağı kodlanarak ilan edilir. DAO'ya katılacak kişiler genellikle bir yönetim token'ine sahip olmalıdır. DAO üyeleri teklif oluşturabilir ve tüm üyeler yönetim token'leri ile oy kullanarak kararları belirler. Belirlenen oy oranına ulaşıldığında akıllı sözleşmeler kararları otomatik olarak uygulamaktadır.

Örneğin; MakerDAO, DAI stablecoin'ini yöneten en büyük DeFi DAO'larından biridir. Kullanıcılar DAI'nin para politikasına oy verebilir. DAI'nin faiz oranları, akıllı sözleşmelerle topluluk tarafından belirlenir.

5.6. Sağlık Hizmetleri

Blok zinciri teknolojisi sağlık alanında daha güvenli, şeffaf, hasta odaklı ve verimli bir sistemin altyapısını kurma vaadi taşımaktadır. Ancak bu gelişmelerin oldukça zaman alacağı kanaatindeyiz. Yaptığımız incelemelerde blok zinciri teknolojisinin medyada çok yer aldığı yıllarda sağlık alanında yapılan projelerin oldukça kısa ömürlü olduğu ve devam eden projelerin birçoğunun da sonradan terk edildiği görülmektedir. Sağlık sektöründe verilerin hassas doğası, regülasyonlar ve geleneksel sağlık sisteminin yavaş yapısının bu projelerin gelişmesini engellemiş olabileceği değerlendirilmektedir.

Bu konuda verebileceğimiz en başarılı uygulama Estonya sağlık sisteminde blok zinciri kullanımıdır. Estonya, sağlık sistemindeki veri güvenliği sorunlarına çözüm olarak KSI Blockchain (Keyless Signature Infrastructure) adı verilen bir blok zinciri teknolojisini uygulamaya koymuştur. Bu sistem, Guardtime [88] adlı Estonya merkezli bir teknoloji firması tarafından geliştirilmiştir. Bu sistemde veriler şifrelenmemekte, bunun yerine her veri bloğuna zaman damgalı bir dijital imza atanmaktadır. Veriler merkezi bir otoriteye

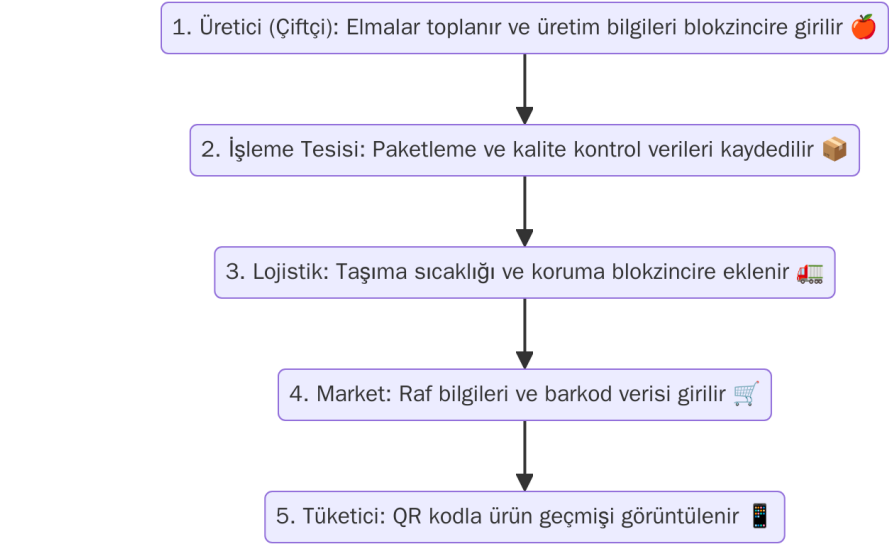
ihtiyaç duymadan değıştirilemez şekilde saklanmaktadır. Hangi veriye, ne zaman ve kim tarafından erişildiğı sistem tarafından kaydedilmektedir. Bu teknoloji sayesinde, herhangi bir hastane çalışanı ya da dış sistem veriler üzerinde oynama yapamamaktadır. Hastaların sağlık verilerinin hem gizliliğı hem de bütünlüğü korunmaktadır. Veri güvenliğı anlamında önleyici değil tespit edici yaklaşım sergilenmektedir.

5.7. Tedarik Zinciri Yönetimi

Tedarik zinciri yönetimi blok zinciri teknolojisinin belki de en kullanılabılır olduğı saha gibi durmaktadır. İzlenebilirlik, şeffaflık ve güvenilirlik, otomasyon ve verimlilik gibi blok zincirinin temel kavramları tedarik zincirinin ihtiyaç duyduğu kavramlardır. Bu açıdan bakıldığında incelemelerimizde bu konuda dünya çapında isim yapmış projeler görmeyi bekliyorduk. Ancak bu konuda da gerçek hayat uygulamaları ile dijital dünya uygulamalarının iletişime geçmesi problemi göze çarpmaktadır. Bir başka husus da sisteme bütün paydaşların (çiftçi, üretici, lojistikçi, dağıtıcı, market, tüketici vs.) girmesi gerekirken bazı paydaşlar bu teknolojiyi kullanacak bilgiye sahip değilken bazı paydaşlar bilgilerini paylaşmak istememektedir. Firmaların kullandığı birbirinden farklı sistemlerin blok zincirine entegrasyonu ciddi bir sıkıntı yaratmaktadır. Blok zincirinin her başlığında gündemimize giren hukuki belirsizlikler burada da projelerin gelişmesini engellemektedir. Projeler teknik olarak hazır olsa da bu konuda da ekosistemin hazır olmaması en büyük sorun olarak karşımıza çıkmaktadır.

Uygulamalar dünya çapında incelendiğinden farklı coğrafyalarda uygulanan yerel projelerin hepsi kapsama alınıp incelenmemiştir. Ancak global anlamda en bilindik proje olduğunu düşündüğümüz IBM Food Trust [89] projesi incelenmiştir. Proje yiyeceklerin üretimden market rafına gelinceye kadar geçirdikleri süreci blok zincirine kaydetmekte ve QR kod ile ürün üzerinden ürün geçmişini görebilmeyi sağlamaktadır.

Şekil 26: IBM FoodTrust Projesi



İlk dikkatimizi çeken husus IBM’in sistemi kullanan tedarikçi sayıları ve işin finansal boyutu hakkında bilgi paylaşıyor olmasıdır. İnternet üzerinden yapılan aramalarda 2023 yıllarında oldukça popüler olan projenin daha sonra popüleritesini kaybettiği görülmektedir. Burada sistemlerin veri doğruluğu sağlayacak mekanizmaları kurması gerekir. Örneğin bir meyve hasatı sonrası çiftçinin girdiği veri doğru kabul edilerek sistem çalışacaktır. Burada çiftçinin elinde kalan başka malları taze olarak sisteme girmesinin önünde bir engel yoktur. Zincirin her halkasında verinin doğruluğunu kontrol eden mekanizmalar olmak zorundadır. Ayrıca her sektörün bu sisteme uyumlu sistemler kullanması ve teknolojik okur yazarlık bilmesi ya da bu eğitimleri alması gerekecektir. Bu da oldukça maliyetli bir süreç olarak karşımıza çıkmaktadır.

5.8. Kamu Hizmetleri ve Yönetim

Blok zincirinin merkeziyetsizlik felsefesiyle yola çıkmasına karşın, arkasında merkezi bir otorite olan projelerin daha başarılı ve sürdürülebilir olduğu görülmektedir. Burada birkaç husus öne çıkmaktadır. Merkezi otoriteler yasal yetkilerini kullanarak sorunları giderebilir, veri giriş, doğrulama yapabilir ve zorunluluk koyabilirler. Merkezi otoriteler uzun süreli fonlama yapabilmektedirler. Vatandaş sistemi kullanmaya zorlayabilirler. Birçok insan kamu kurumlarına hala blok zinciri sistemlerinden daha fazla güvenmektedir.

Estonya da e-voting (dijital oylama) kullanılmaktadır. Seçim zamanı isteyen Estonyalı sandık başına giderken isteyen Estonya vatandaşı bilgisayar ya da cep telefonu üzerinden blok zinciri teknolojisi ile oy kullanabilmektedir. 2023 seçimlerinde katılımcıların %51 e-voting kullanmıştır [90]. Bir başka örneğimiz Gürcistan'ın Tapu ve Mülkiyet Sisteminde blok zinciri kullanımı. Bu sistemde aslında işlemler hukuki zorunluluklar sebebiyle merkezi veritabanına yazılmaktadır. Ancak bu verinin hash'i Bitcoin blok zincirine yazılıyor. Böylelikle devletin veritabanında değişiklik yapılmaya çalışılsa bile bu tespit edilebiliyor.

Bir başka örnek olarak Çin'in Shenzhen Belediyesinin et ve sebze izleme sistemi verilebilir. Aslında IBM Food Trust ile aynı mantıkta çalışan bu sistemi belediye lokal ve dar bir çevrede işler bir hale getirmiştir [91].

6. BLOK ZİNCİRİ VE KRİPTO PARALARIN GELECEĞİ

Blok zinciri teknolojisinin ortaya çıkışından bu yana potansiyelini yakalayamadığını görmekteyiz. Yapılan projeler ve araştırmalar çok farklı alanlarda yenilikçi çözümler sunmakla birlikte tam anlamıyla toplum tarafından kabul görerek yaygın bir gerçek hayat uygulamasına dönüşmemektedir. Bu konuda en başarılı uygulama olan kripto paralar diğer uygulamalara göre oldukça ön plana çıkmaktadır. Akıllı sözleşmeler ise yine kripto paralarla ilgili projelerde oldukça yaygın ve etkin kullanılmaktadır. Kripto paralarda da dönemsel olarak ilginin artması ve azalmasına bağlı olarak yeni teknolojiler ve yeni projeler ortaya konmaktadır. Blok zinciri teknolojisinin belirli bir olgunluğa geldiği, birçok yeni teknoloji ve yöntemi ortaya koyduğu, ve gerçek hayat uygulamalarına alternatif çözümler sunduğu kanaatindeyiz. Buradan sonra zamanla insanların bu ekosistemi anlaması ve kabullenmesi bir süre sonra büyük oranda gerçekleşecektir. Bu süreçlerin tamamlanması sonrasında blok zinciri teknolojisinin potansiyelini tam olarak ortaya koyabileceğini değerlendiriyoruz.

İnsanların bu ekosisteme temkinli yaklaşmasının altında kripto dünyasında yaşanan siber saldırı ve dolandırıcılık olayları, teknolojisinin güvenli olmasına karşın son kullanıcılar için bir üçüncü taraf hizmeti almadan kullanılamayacak karmaşıklıkta olması, merkezi otoritelerin güvenlik kaygıları ile sürece yaklaşması ve yasal düzenleme eksiklikleri olduğu kanaatindeyiz. Blok zinciri teknolojisinin yaygınlaşma sürecini farklı başlıklar altında inceleyelim.

6.1. ETF (Exchange Traded Fund – Borsa Yatırım Fonu)

Borsa Yatırım Fonları (ETF), birden fazla menkul kıymeti (örneğin hisse senetleri, tahviller, emtialar, endeksler) tek bir yatırım aracı altında toplayan, ancak hisse senedi gibi borsada alınıp satılabilen fonlardır. Bu yapılar, yatırımcılara çeşitlendirme imkânı sunarken, işlem kolaylığı ve likidite avantajı da sağlamaktadırlar.

Bu konunun tezimiz ile bağlantısı, son yıllarda bazı fonların Amerika Birleşik Devletleri Menkul Kıymetler ve Borsa Komisyonu'na (SEC) başvurarak kripto varlık tabanlı ETF'ler, özellikle Bitcoin ETF'leri oluşturmak istemeleriyle kurulmaktadır. Uzun süre tartışılan bu

başvuruların ardından Ocak 2024'te SEC, 11 adet spot Bitcoin ETF'ine onay vermiştir [92]. Ardından, Mayıs 2024'te 8 adet Ethereum ETF'i de SEC tarafından onaylanmıştır. Bu gelişmeler, kripto paraların daha geniş yatırımcı kitleleriyle buluşmasının önünü açmıştır.

Ancak burada daha önemli olan nokta, emeklilik fonları, sigorta şirketleri ve büyük kurumsal yatırımcıların, bu ETF'ler aracılığıyla kripto varlıklara doğrudan yatırım yapabilme imkânına kavuşmuş olmalarıdır. Geleneksel finans sisteminin bu en büyük oyuncularının, regülasyona tabi borsa yatırım fonları aracılığıyla Bitcoin ve Ethereum'a yönelmeleri, bu varlıkların bir anlamda finansal sistem tarafından tanınması ve güvenilir bulunması olarak da yorumlanabilir. Bu da blok zinciri teknolojisinin toplumun geniş kesimleri tarafından kabul görmesini hızlandırıcı bir etki yapmaktadır.

6.2. Metaverse

Metaverse, insanların dijital avatarlarla 3D sanal dünyalarda ya da artırılmış gerçeklik ortamlarında etkileşime geçtiği; ürün ve hizmet alışverişi yaptığı dijital bir evrendir. Özellikle pandemi döneminde, fiziksel dünyanın kısıtlı hale gelmesiyle birlikte geleceğin teknolojisi olarak büyük ilgi görmüştür. Bu süreçte birçok yatırımcı metaverse platformlarından dijital arsa satın almış, bankalar sanallaşmış şubeler açmış ve bazı sanatçılar metaverse ortamında konserler vermiştir.

Ancak kısa sürede popülerleşen bu teknoloji, benimsenecek düzeyde kullanıcı deneyimi sunamamış ve zamanla ilgi kaybına uğramıştır. Bu düşüşün temel nedenleri arasında; grafik kalitesinin yetersizliği, VR gözlüklerin hantal ve konforsuz oluşu, internet altyapısının gerçek zamanlı etkileşim için elverişsiz olması gibi teknik sınırlamalar yer almaktadır. Ayrıca, metaverse içeriğinin sınırlı olması ve ekonomik fayda vadeden projelerin çoğunun beklentileri karşılamaması da kullanıcıların ilgisini azaltmıştır.

Bununla birlikte, metaverse kavramının tamamen yok olduğunu söylemek yanıltıcı olur. Aksine, bu alanın gelecekte daha gelişmiş altyapı ve daha entegre teknolojilerle yeniden gündeme gelmesi muhtemeldir. Bu noktada önemli olan, metaverse'ün yalnızca görsel ya da

eğlence temelli değil, aynı zamanda mülkiyet, ekonomi, etkileşim ve kimlik gibi temel dijital yapı taşlarını barındırmasıdır.

İşte bu noktada blok zinciri teknolojisi, metaverse'ün en güçlü altyapı bileşeni olarak öne çıkmaktadır. Çünkü dijital varlıkların sahipliliği için NFT'ler, metaverse ekonomisinin inşası için kripto paralar, güvenli dijital ödemeler için kripto cüzdanlar ve DeFi uygulamaları, işlemlerin şeffaflığı ve denetlenebilirliği için blok zincir kayıtları, kullanıcılar arasında güvenli işlem yapılabilmesi için akıllı sözleşmeler, farklı platformlar arasında veri ve varlık geçişini sağlamak için ise zincirler arası birlikte çalışabilirlik (interoperability) gibi çözümler doğrudan blok zinciriyle ilişkilidir.

Dolayısıyla, gelecekte metaverse'ün gelişmesi için sadece donanım ve altyapı değil, aynı zamanda blok zinciri temelli bir ekonomik ve yönetim altyapısının varlığı da kritik olacaktır. Bu entegrasyon, yalnızca kullanıcı deneyimini güçlendirmekle kalmayacak, aynı zamanda kullanıcıların mülkiyet hakkı, gelir elde etme potansiyeli ve dijital kimlik yönetimi gibi alanlarda da güven inşa edecektir.

Sonuç olarak, metaverse kavramı, bugünkü anlamıyla değilse de, gelecekte daha sade, işlevsel ve entegre biçimlerde hayatımıza yeniden girebilir. Bu bağlamda, metaverse alanında geliştirilecek başarılı uygulamalar, aynı zamanda blok zinciri teknolojisinin en kapsamlı test ortamlarını oluşturacak ve toplum nezdinde bu teknolojiye duyulan güvenin artmasına katkı sağlayacaktır.

6.3. Web 3.0

Web 3.0, merkeziyetsiz ve verinin kontrolünün kullanıcılarda olduğu, blok zinciri temelli yeni nesil bir internet anlayışıdır. Bu yapının tam anlamıyla hayata geçirilebilmesi için bazı temel bileşenlerin varlığı gereklidir. Dijital içeriklere sahipliğin sağlanabilmesi için NFT'ler ve kripto cüzdanlar, güvenliğin artırılması için merkezi olmayan dağıtık veri sistemleri, ödeme sistemleri için kripto paralar ve DeFi (merkeziyetsiz finans) uygulamaları, karar alma mekanizmaları ve yönetim için DAO'lar (merkeziyetsiz otonom organizasyonlar) ve güvenli etkileşim için ise akıllı sözleşmeler bu bileşenler arasında yer alır.

Ancak, bu potansiyeline rağmen Web 3.0 uygulamalarının yaygınlaşması ve kullanışlı hale gelmesi ciddi teknik zorluklarla karşı karşıyadır. Bunların başında ölçeklenebilirlik sorunu gelmektedir. Blok zincirlerin mevcut durumda yüksek kullanıcı sayılarını ve büyük veri hacimlerini yönetme kapasiteleri sınırlıdır. Ayrıca, Web 3.0'ın temel prensiplerinden biri olan merkeziyetsizlik, beraberinde içerik kontrolü ve moderasyon sorunlarını da getirmektedir. Özellikle uygunsuz ya da yasa dışı içeriklerin tespiti ve ayrıştırılması, merkezi olmayan sistemlerde karmaşık bir problem olarak karşımıza çıkmaktadır.

Öte yandan, Web 3.0 ve blok zinciri teknolojilerinin birbirlerini tamamlayan ve güçlendiren iki unsur olarak gelişeceğini düşünmekteyiz. Blok zinciri teknolojilerinde yaşanacak herhangi bir gelişme, Web 3.0 ekosisteminin performansını, ölçeklenebilirliğini ve kullanılabilirliğini doğrudan artıracaktır. Benzer şekilde, Web 3.0 tarafında kullanıcı deneyimi, dijital kimlik çözümleri ve birlikte çalışabilirlik gibi alanlarda kaydedilen ilerlemeler de blok zinciri teknolojilerinin benimsenmesini hızlandıracaktır.

6.4. IoT (Internet of Things - Nesnelerin İnterneti)

IoT fiziksel cihazların (sensörler, araçlar, ev aletleri vb.) internete bağlanarak birbirleriyle ya da merkezi bir sistemle veri alışverişi yaptığı ve böylece etkileşime girdiği bir teknolojidir. Cihazlar yalnızca merkezi bir sistemle değil, kendi aralarında da doğrudan iletişim kurarak veri paylaşabilirler. Tezimiz kapsamında değindiğimiz birçok teknolojinin belirli dönemlerde popüler olup ardından duraklama dönemine girdiğini, hatta bazılarının tamamen ortadan kaybolduğunu görmekteyiz. Buna karşılık IoT, gündeme geldiği ilk günden itibaren kesintisiz biçimde yükselen bir gelişim grafiği sergilemektedir.

Blok zinciri ve IoT ilişkisini değerlendiren kaynaklarda genellikle blok zincirinin IoT'ye sağlayacağı avantajlar öne çıkarılmaktadır. Örneğin, blok zinciri sayesinde IoT sistemlerinde veri güvenliği ve değiştirilemezlik sağlanabilecektir. Merkezi sistemlerin oluşturduğu risklere karşı, merkeziyetsiz blok zinciri yapısı sayesinde tek nokta kaynaklı hata ve saldırılar azalacaktır. Ayrıca blok zincirindeki veriler şeffaf, denetlenebilir ve güvenilir olacaktır. Buna ek olarak, akıllı sözleşmeler aracılığıyla blok zinciri teknolojisi,

makineler arası otomatik ödeme ve işlemleri mümkün kılarak IoT'nin ekonomik yönünü güçlendirecektir.

Bu görüşlere katılmakla birlikte, biz bu ilişkinin karşılıklı olduğunu ve IoT'nin de blok zincirine çok daha büyük ve anlamlı katkılar sağlayabileceğini düşünmekteyiz. Blok zincir teknolojisi, gerçek hayat ile bağlantı kurduğu noktalarda ciddi bir sorun olan doğru, güvenilir ve gerçek zamanlı veri sağlama problemini yaşamaktadır. Blok zincirinde "oracle" olarak adlandırılan, zincir dışından veri getiren sistemlerin gelecekteki baş aktörleri olarak IoT cihazları öne çıkacaktır. Bu cihazlar arasında makineler, sensörler, araçlar ve hatta robotlar bulunacaktır. Örneğin tedarik zinciri uygulamalarında, sisteme girilen meyvenin tazeliğini ve kalitesini sensörlerle otomatik ve gerçek zamanlı olarak kontrol eden bir IoT sistemi, mevcut manuel ve hataya açık yöntemlerden çok daha kullanışlı, güvenilir ve hızlı olacaktır.

Bu bağlamda IoT'nin gelişiminin, blok zinciri teknolojisinin yaygınlaşmasında ve gerçek hayatla temas ettiği alanlarda uygulamalarının güçlenmesinde oldukça önemli ve etkili bir rol oynayacağı sonucuna varıyoruz. Bu iki teknoloji arasındaki ilişkinin karşılıklı olarak güçlendirme etkisine sahip olduğunu düşünmekteyiz.

6.5. Yapay Zekâ

Kanaatimizce blok zinciri teknolojisinin hem popülaritesini azaltan hem de uzun vadede onu daha güçlü hâle getirecek olan temel faktör, yapay zekâ teknolojisidir. Tez çalışmalarımıza başladığımız 2023 yılının son çeyreğinde yapay zekâ oldukça gündemdeyken, blok zinciri teknolojisiyle ilgili de yeni gelişmeler sıklıkla duyulmaktaydı. Ancak zamanla yapay zekânın gündemi tamamen ele geçirdiğini ve blok zinciriyle ilgili haberlerin giderek azaldığını, hatta son zamanlarda neredeyse tamamen durağan bir seyir izlemeye başladığını gözlemlemekteyiz.

Bu durumun temel nedenlerinden biri olarak, büyük yatırımcıların ilgisinin hızla yapay zekâ teknolojilerine kaymış olmasını gösterebiliriz. Örneğin, Microsoft'un OpenAI'a yaptığı milyarlarca dolarlık yatırım ya da Google'ın Gemini ve Bard gibi yapay zekâ modellerine

yönelik büyük bütçeli çalışmaları, bu eğilimi net biçimde göstermektedir. Yatırımların yapay zekâya kayması, blok zinciri alanındaki projelerin finansal destek bulmasını zorlaştırmıştır.

Diğer bir önemli husus ise blok zinciri teknolojisinin belli bir kavramsal doygunluğa ulaşmasıdır. Merkeziyetsizlik, akıllı sözleşmeler ve DAO'lar gibi temel kavramlar olgunlaşmış, sistem arkasındaki kriptografik mekanizmalar etkinliğini ispatlamış ve birçok başarılı proje hayata geçirilmiştir. Artık bu teknolojinin toplum tarafından kabul edilmesi ve yaygınlaşması için gereken zamanın geçmesi beklenmektedir.

Buna karşılık yapay zekâ, hızla gelişmekte ve metin tabanlı sohbet eden modellerden, kullanıcılar adına gerçek dünyada aksiyon alabilen, görev gerçekleştirebilen otonom modellere dönüşmektedir. Bu dönüşüm, kanaatimizce blok zinciri teknolojisinin yaşadığı durgunluğa son verebilecek bir potansiyel taşımaktadır. Yapay zekânın blok zinciri üzerindeki etkisi düşünüldüğünde, özellikle blok zincirindeki merkeziyetsizliği tehdit eden yazılım güncelleme kararları yapay zekâ tarafından daha şeffaf, hızlı ve objektif bir biçimde alınabilir.

Benzer şekilde, blok zinciri ekosisteminde sıkça karşılaşılan akıllı sözleşmelerdeki güvenlik açıkları, bugüne kadar büyük maddi kayıplara yol açmıştır. Bu tür hataların ve açıkların tespit edilmesinde yapay zekânın aktif olarak kullanılması, sistemin güvenilirliğini önemli ölçüde artıracaktır. Ayrıca blok zinciri teknolojisinin ortalama kullanıcı açısından çok karmaşık olması, kullanıcıları üçüncü taraf yazılımlara bağımlı kılmaktadır. Bu durum, kullanıcı dostu ve kişiselleştirilmiş yapay zekâ asistanlarının yaygınlaşmasıyla ortadan kalkabilir. Kullanıcılar, yapay zekâ desteğiyle kendi özel akıllı sözleşmelerini rahatlıkla oluşturabilir ve kişisel kripto varlıklarını üçüncü taraflara ihtiyaç duymadan güvenle yönetebilirler.

Yapay zekânın blok zinciri teknolojisinin kalbine yerleştiği yeni bir mimarinin, kullanıcıların teknolojiye güvenini hızla artıracak ve blok zincirinin toplumda yaygın olarak kabul edilmesini önemli ölçüde hızlandıracağını düşünüyoruz.

6.6. Yasal Düzenlemeler

Blok zinciri teknolojisinin en temel problemi, aynı zamanda temel vaadi olan merkeziyetsizlik kavramıdır. Merkeziyetsizlik kavramı toplumun önemli bir bölümü tarafından "kontROLSÜZ", "güvensiz" ve "hukuksuz" olarak algılanmaktadır. Bu algının oluşmasında, blok zinciri tabanlı sistemlerde yaşanan olumsuz olayların medyada geniş yankı bulması büyük rol oynamaktadır. Örneğin; siber saldırı olaylarında başvurulabilecek bir merkezi otoritenin olmaması, bazı merkeziyetli kripto projelerinde (örneğin Terra/Luna veya FTX borsası örneğinde olduğu gibi) sistemin kontrolünü elinde tutan kişilerin ya da ekiplerin şeffaf olmayan kararlarla kullanıcıları mağdur edebilmesi, toplumda blok zinciri teknolojisine yönelik şüpheleri artırmaktadır.

Sistem yaratıcılarının, konsensüs algoritmaları, elektronik imzalar ve hash algoritmaları gibi önleyici güvenlik önlemlerini titizlikle tasarımlarına rağmen, bu önlemler ortalama kullanıcıların anlayabileceğinden çok daha karmaşıktır. Ayrıca blok zinciri sistemlerinin güvenlik yaklaşımında genellikle sadece "önleyici" mekanizmalar kullanılır. Buna karşılık, günümüzde merkezi otoriteler önleyici, tespit edici ve hesap sorucu yöntemleri bir arada kullanırlar. Örneğin banka hesaplarından gerçekleşen dolandırıcılık olaylarında, sadece önleyici güvenlik tedbirleri değil, çalınan paranın takibi (tespit edici mekanizma) ve yasal süreçler aracılığıyla geri alınması (hesap sorucu mekanizma) mümkün olmaktadır. Blok zinciri sistemleri ise genellikle bu tür tespit edici ve hesap sorucu mekanizmalara sahip değildir. Bu durum, toplumun bu teknolojiyi kabulünü güçleştirmektedir.

Ancak, son yıllarda yapay zekânın da desteğiyle blok zinciri sistemlerinin bu tür mekanizmaları içermesi konusunda çalışmalar başlamıştır. Yapay zekâ destekli güvenlik sistemleri, blok zinciri üzerinde gerçekleşen anormal işlemleri gerçek zamanlı olarak tespit edebilir, ilgili tarafları uyarabilir ve gerekli önlemleri alabilir. Bu tür teknolojiler, blok zinciri ekosisteminde yaşanan problemlerin azaltılmasına ve kullanıcı güveninin artmasına önemli katkılar sunabilir.

Blok zinciri teknolojisi için doğrudan düzenleyici bir çerçeve oluşturmak zordur, çünkü teknoloji nötr ve genel amaçlıdır. Düzenleyici kurumlar, doğrudan teknoloji yerine, onun

kullanım alanlarına yönelik düzenlemeler geliştirir. Bu sebeple günümüzde kripto paralar ve DeFi gibi öncü alanlar üzerinden düzenlemeler yapılmaktadır.

Tez çalışmamızda ülke uygulamalarının hepsi değerlendirilmemiş bazı öne çıkan düzenlemeler paylaşılmış ve daha sonrasında konu hakkında ülkemizde yaşanan gelişmelere yer verilmiştir. Bu gelişmelerden önemli olanları aşağıda listelenmiştir.

- Amerika Birleşik Devletleri (ABD): ABD, kripto paraları düzenlemek için çeşitli federal ve eyalet düzeyinde yasalar uygulamaktadır. ABD Menkul Kıymetler ve Borsa Komisyonu (SEC), kripto paraların menkul kıymet olup olmadığını belirleyerek düzenlemeler yapmaktadır.
- Avrupa Birliği (AB): AB, kripto varlık hizmet sağlayıcılarının uyum sağlaması gereken kapsamlı bir düzenleme paketi olan MiCA (Markets in Crypto-Assets) üzerinde çalışmaktadır. Bu düzenleme, kripto varlık piyasalarının daha güvenli ve şeffaf olmasını hedeflemektedir.
- Japonya: Japonya, kripto paraları destekleyici bir yaklaşımla yasal düzenlemeler yapmaktadır. Ülkede kripto para borsaları, Finansal Hizmetler Ajansı (FSA) tarafından lisanslanmakta ve denetlenmektedir.
- Singapur: Singapur, blockchain ve kripto paralar için inovasyon merkezleri ve vergi teşvikleri sunarak destekleyici bir yaklaşım benimsemektedir. Ülkede kripto para borsaları, Singapur Para Otoritesi (MAS) tarafından düzenlenmektedir.
- Güney Kore: Güney Kore, kripto paralar için temkinli bir yaklaşım benimsemekte ve mevcut düzenlemeleri adapte etmektedir. Ülkede kripto para borsaları, Finansal Hizmetler Komisyonu (FSC) tarafından denetlenmektedir.

Türkiye’de konu hakkında yapılan yasal düzenlemeleri kronolojik olarak şöyle sıralayabiliriz;

- 25 Kasım 2013 – BDDK’nın İlk Açıklaması: Bankacılık Düzenleme ve Denetleme Kurumu (BDDK), Bitcoin’in elektronik para olarak kabul edilemeyeceği ve mevcut düzenlemelere göre denetlenmesinin mümkün olmadığını vurgulayan bir basın açıklaması yaptı. Bu açıklama, Türkiye’deki ilk resmi kripto varlık yasal düzenlemesi olarak kabul edilmektedir.

- 16 Nisan 2021 – TCMB'nin Ödeme Aracı Olarak Kripto Yasağı: Türkiye Cumhuriyet Merkez Bankası (TCMB), "Ödemelerde Kripto Varlıkların Kullanılmamasına Dair Yönetmelik" yayımladı. Yönetmelikle, kripto varlıkların doğrudan veya dolaylı şekilde ödeme aracı olarak kullanılması yasaklandı. Ödeme ve elektronik para kuruluşlarının kripto platformlarıyla işbirliği yapması engellendi.
- Mayıs 2021 – MASAK'ın Kripto Platformları için Yükümlülük Rehberi: Mali Suçları Araştırma Kurulu (MASAK), kripto varlık hizmet sağlayıcıları (KVHS) için kara para aklama (AML) ve terörün finansmanı (CFT) konusunda ayrıntılı yükümlülük rehberini yayımladı. Kripto platformları MASAK yükümlüsü olarak kabul edildi ve bu platformlara KYC/AML politikaları uygulama ve şüpheli işlem bildirimi zorunluluğu getirildi. Temmuz 2021'de MASAK yükümlülükleri, kripto para borsalarında aktif olarak uygulanmaya başladı. Şüpheli işlemlerin takibi ve müşteri bilgilerinin toplanması zorunlu hale geldi.
- 2 Temmuz 2024 – TBMM'de Sermaye Piyasası Kanunu Değişikliği: 7518 sayılı Kanun, 2 Temmuz 2024 tarihinde Resmî Gazete'de yayımlanarak yürürlüğe girdi. Bu kanunla birlikte, kripto varlık hizmet sağlayıcıları (KVHS) ve saklama kuruluşları, Sermaye Piyasası Kurulu (SPK) denetimine tabi tutuldu. KVHS'lerin SPK'dan faaliyet izni (lisans) alması zorunlu hale getirildi. Kanun, kripto varlıkların sermaye piyasası aracı olarak tanımlanmasına ve SPK'nın bu alandaki düzenleyici yetkisinin genişletilmesine olanak tanıdı.
- 8 Ağustos 2024 – SPK'nın Kripto Platformları için İlke Kararları: SPK, i-SPK.35.B sayılı İlke Kararı ile kripto varlık platformlarının kuruluş ve faaliyetlerine ilişkin temel ilkeleri belirledi. Platformların anonim şirket olarak kurulması ve asgari 50 milyon TL sermaye şartı getirildi. Yöneticilerin finansal sektörde deneyimli ve güvenilir kişiler olması gerektiği vurgulandı. Müşteri varlıklarının platformlardan ayrı olarak saklanması ve reklam/promosyon faaliyetlerinin SPK onayına tabi olması gibi düzenlemeler getirildi.

- 13 Mart 2025 – SPK'nın Detaylı Tebliğleri (III-35/B.1 ve III-35/B.2): SPK, III-35/B.1 ve III-35/B.2 sayılı tebliğleri Resmî Gazete'de yayımlayarak yürürlüğe koydu. III-35/B.1 Tebliği, KVHS'lerin kuruluş, faaliyet izni, organizasyon yapısı, iç kontrol sistemleri ve bilgi sistemleri gibi konuları düzenledi. III-35/B.2 Tebliği, KVHS'lerin sunabileceği hizmetler, sermaye yeterliliği, müşteri varlıklarının saklanması ve bağımsız denetim süreçlerine ilişkin esasları belirledi. Tebliğler, KVHS'lerin faaliyetlerini daha şeffaf ve güvenilir bir şekilde yürütmelerini sağlamayı amaçladı.
- Mayıs 2025 – TÜBİTAK BİLGEM'in Teknik Standartları: TÜBİTAK BİLGEM, KVHS'lerin bilgi sistemleri ve teknolojik altyapılarına ilişkin teknik kriterleri belirleyen bir doküman yayımladı. Doküman, platformların siber güvenlik, veri koruma, işlem bütünlüğü ve sistem denetimi gibi konularda uyması gereken standartları içeriyor. Bu kriterlere uyum, SPK lisanslama sürecinde zorunlu tutuldu.
- 9 Mayıs 2025 – SPK'nın Rezerv Kanıtı Düzenlemesi: SPK, kripto varlık platformlarının müşteri varlıklarını koruması amacıyla rezerv kanıtı (Proof-of-Reserve) denetim yükümlülüğü getirdi. Platformlar, müşteri varlıklarının doğruluğunu ve varlığını belgeleyen rezerv kanıtı denetimlerini bağımsız denetim kuruluşlarına yaptırmak zorunda olacak. Aynı denetim şirketine arka arkaya en fazla 12 dönem denetim yaptırılabilir ve bir başdenetçi aynı platform için en fazla 4 dönem denetim raporu imzalayabilecek. Denetim raporları, denetim döneminin bitimini takip eden 10 iş günü içinde platformun yönetim kuruluna teslim edilecek.

7. TOKENİZE ÖNERİSİ

Bu bölümde tezin genelinde anlatılan konular çerçevesinde ve konunun gerçek bir uygulama üzerinde olumlu ve olumsuz yönlerini, yarattığı fırsatları ve zorluklarını ortaya koyacak bir model önerilmektedir. Model kurgulanırken mümkün olan en merkeziyetsiz, yatırımcı ve müşteri haklarını koruyan bir yaklaşım seçilmiştir. Şirketin kar etmesi ve sistemin devamlılığı göz önünde bulundurulmuş ancak odak noktası olarak kar maksimizasyonu alınmamıştır. Burada olabilecek en şeffaf ve güvenli yapı kurgulanırken bazı noktalarda karlılık ikinci plana bırakılmıştır.

7.1. Sektör seçimi

Öncelikle örnek olarak seçilecek sektörün belirlenmesinde araç kiralama işlemlerinin ülkemizde standart bir uygulama haline gelmeye başladığı, bazı yatırımcıların araç satın alarak bu şirketlere kiraladığı, araç kiralama fiyatlarının yüksekliği, bu alan da hem araç kiralayan müşterilerin hem de araç sahiplerinin araç bilgilerine ulaşmadaki kısıtları, araç kiralamada kredi kartı talep edilmesi gibi hususlar dikkate alınmıştır.

Bu çerçevede; araç kiralama yatırımına erişimleri kolaylaştırmak, tokenizasyon ile yatırım çeşitliliğini artırmak, finansal süreçlerde şeffaflığı ve güvenilirliği blok zinciri teknolojisi ile sağlamak, merkeziyetsiz bir organizasyon (DAO) yapısıyla demokratik bir yönetim gerçekleştirmek, yüksek likidite ve esneklik sunarak yatırımcı memnuniyetini artırmak için “Araba Kiralama” isimli araç kiralama işlemleri yapan bir şirketin kurulacaktır.

7.2. Platform seçimi

Şirket bir blok zinciri üzerinde çıkarılan tokenler ve akıllı sözleşmelerle süreçlerini yönetecektir. Bu kapsamda şirketin planlanan bütün blok zinciri tabanlı işlemlerini yürütebilecek platform olarak Avalanche seçilmiştir. Avalanche (AVAX), merkeziyetsiz uygulamalar (dApps) ve akıllı kontratlar geliştirmek için yüksek performanslı, ölçeklenebilir, hızlı ve düşük maliyetli blok zinciri platformudur. Ethereum uyumludur ve Solidity tabanlı akıllı kontratları desteklemektedir.

Böylelikle; araç yatırım süreçlerinin hızlı ve düşük maliyetle gerçekleşmesi, token transferleri ve ödeme işlemlerinin anlık ve ekonomik olması, Güvenli ve şeffaf veri yönetimi, blok zincir üzerinde değişmez kayıtların tutulması ile merkeziyetsiz yönetim ve otomasyonun etkin şekilde sağlanabilmesi planlanmaktadır.

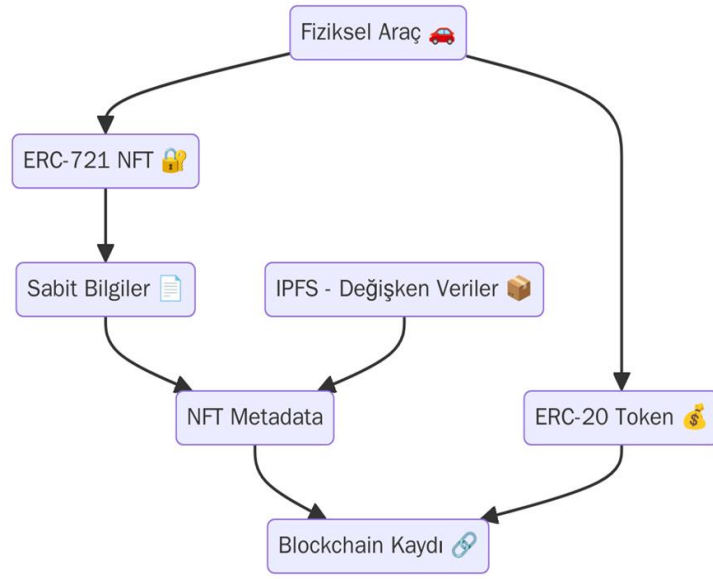
Proje, Avalanche'ın Contract Chain (C-Chain) altyapısını kullanacaktır. C-Chain, Ethereum Virtual Machine (EVM) uyumludur ve akıllı sözleşmelerin sorunsuz çalışmasını sağlar. Tüm araç tokenleri, NFT'ler, ödeme sistemleri ve DAO yönetimi Avalanche'ın C-Chain ağında konumlandırılacaktır. Araç sahipliğini temsil eden tokenler için ERC-20 standardı, Araç bilgilerini tutmak için ERC-721 (NFT) standardı kullanılacaktır.

7.3. Araçların tokenizasyonu

Tokenizasyon, gerçek dünya varlıklarının dijital tokenlere dönüştürülmesi sürecidir. Araç kiralama projesinde, araçların mülkiyet hakları dijital tokenlerle temsil edilecektir. Proje kapsamında her araç için ayrı ERC-20 tokenleri oluşturulacaktır. Token sembolü olarak aracın plakası kullanılacaktır. Böylece tokenler birbirleriyle karışmayacak ve her aracın sahipliği net biçimde belirlenecektir. ERC-20 tokenleri kolay transfer edilebilirlik, yüksek likidite ve çoklu cüzdan desteği avantajlarına sahip olacaktır.

NFT'ler (Non-Fungible Token), blockchain teknolojisi üzerinde oluşturulan benzersiz ve değiştirilemez dijital varlıklardır ve NFT'ler sayesinde fiziksel veya dijital varlıkların sahipliği ve özgünlüğü şeffaf bir şekilde doğrulanabilmektedir. Araç kiralama projesinde, her araç için Avalanche ağı üzerinde ERC-721 standartlarında benzersiz bir NFT oluşturulacaktır. NFT'ler her araç için dijital bir kimlik olarak kullanılacak, araç plakası, araç markası ve modeli, üretim yılı, araç ilk kayıt tarihi gibi sabit bilgiler kaydedilecektir.

Şekil 27: Araçların tokenleştirilmesi



Ancak aracın değişken bilgileri (hasar, bakım kayıtları, kilometresi vb.) bilgileri NFT meta datasında tutulamamaktadır. Bu tür sürekli değişen veriler IPFS (InterPlanetary File System) gibi merkeziyetsiz bir veri depolama platformunda tutulacaktır. NFT'nin blok zinciri üzerindeki metadata alanında, IPFS'deki araç dosyalarının bağlantısı bulunacak araç verileri güncellendiğinde IPFS üzerindeki dosya güncellenerek, NFT'nin metadatasındaki IPFS bağlantısı değiştirilecektir. Böylelikle blok zinciri üzerinde veri saklama maliyeti düşürülecek, büyük veri dosyaları saklanabilecek, verilerin doğrulanabilir ve kalıcı olması sağlanacaktır.

Araçların tokenleştirilmesi sürecinde oluşturulacak akıllı sözleşme için aşağıdaki tabloda yer alan yol haritasının kullanılması faydalı olacaktır.

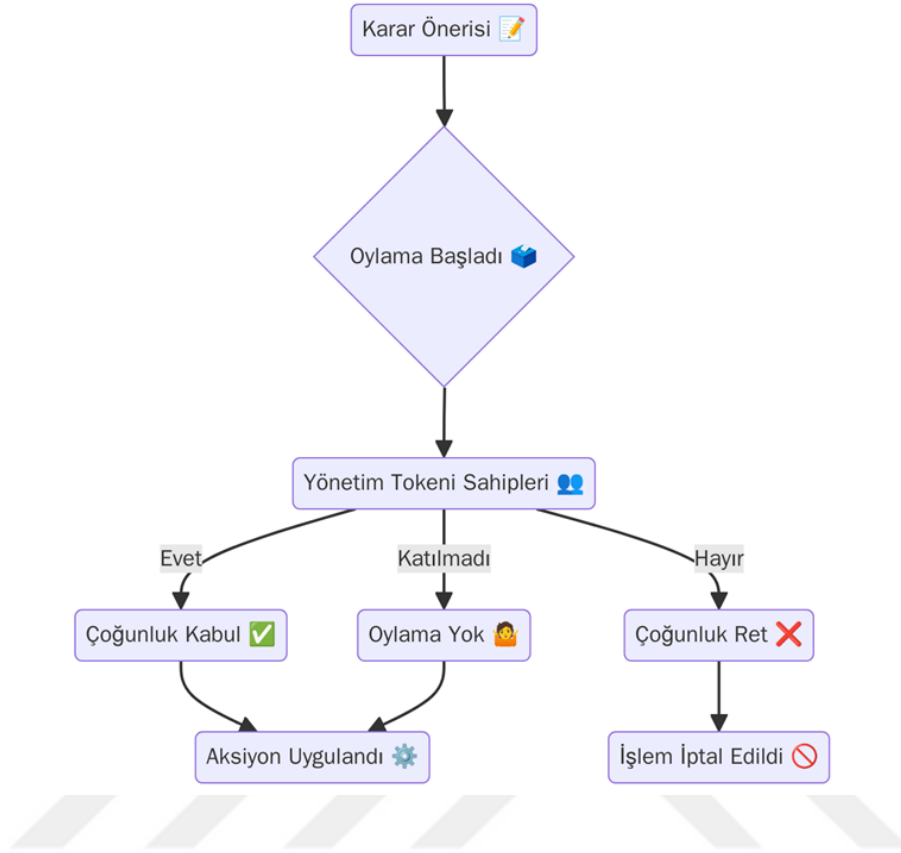
Tablo 12: Araçların tokenleştirme süreci kodlama yol haritası

	Fonksiyon / İşlem	Açıklama	Girdi Parametreleri	Çıktı Değerleri	Güvenlik Şartları	Bağlantılı İşlem / Trigger
1	aracBilgisiKaydet	Araç bilgileri sisteme kaydedilir (model, marka, yıl, şasi numarası gibi).	Araç kimliği, model, marka, yıl, özellikler	Araç kayıt verisi oluşturulur	Yalnızca yetkili kişi (admin) çağırabilir	Yönetici yeni araç ekleyince
2	tokenlestirmeBaslat	Belirtilen araç için tokenleştirme işlemi başlatılır. Toplam token adedi belirlenir (örn. 1000 adet).	Araç kimliği, token adedi	Tokenleştirme kaydı oluşturulur	Araç daha önce tokenleştirilmemiş olmalı	aracBilgisiKaydet sonrası
3	tokenSembolOlustur	Araca özel bir token sembolü ve adı oluşturulur (örnek: 34ABC123_TKN).	Araç kimliği	Token adı ve sembolü atanır	Sembol eşsiz olmalı	tokenlestirme Baslat içinde tetiklenir
4	tokenlariMintEt	Araç için toplam belirlenen miktarda token üretilir (mint edilir) ve şirket cüzdanına aktarılır.	Token adedi	Tokenlar oluşturulur ve şirkete atanır	Token adedi sistem parametreleri içinde olmalı (örn. sabit 1000)	tokenSembolOlustur sonrası
5	tokenlariKitle	Üretilen tokenlar yatırım turu başlatılana kadar transfer edilemez şekilde kilitlenir.	Token adresi	Token transferi kısıtlanır	Yatırım turu başlamadan önce işlem yapılmamalı	tokenlariMint Et sonrası
6	yatirimTuruneHazirla	Tokenlar yatırım fonlama turu için serbest bırakılır, yatırımcılara satışa hazır hale gelir.	Token adresi	Token durumu "yatırıma açık" yapılır	Araç bilgisi ve tokenler eksiksiz tanımlanmış olmalı	Fonlama başlatılmadan hemen önce

7.4. Merkeziyetsiz Organizasyon Yapısı (DAO)

DAO (Decentralized Autonomous Organization), tüm karar süreçlerini blok zinciri üzerinde kayıtlı akıllı sözleşmeler aracılığıyla otomatik ve şeffaf biçimde yürüten merkeziyetsiz yönetim yapılarıdır. Araç kiralama projesinde projeye özel bir Yönetim (Governance) token çıkarılacaktır. DAO yapısı, yönetim kararlarının demokratik ve merkeziyetsiz şekilde alınmasını sağlayacaktır. Yönetim Tokeni sahipleri, araç alımı, satışı, sigorta işlemleri, bakım süreçleri ve diğer önemli konular hakkında DAO aracılığıyla oy kullanacak ve kararlar böyle verilecektir.

Şekil 28: Merkeziyetsiz Organizasyon karar süreci



Merkeziyetsiz organizasyon yapısı sürecinde oluşturulacak akıllı sözleşme için aşağıdaki tabloda yer alan yol haritasının kullanılması faydalı olacaktır.

Tablo 13: Merkeziyetsiz organizasyon yapısı kodlama yol haritası

	Fonksiyon / İşlem	Açıklama	Girdi Parametreleri	Çıktı Değerleri	Güvenlik Şartları	Bağlantılı İşlem / Trigger
1	yonetimOnerisiOlustur	Token sahipleri sistem üzerinde bir değişiklik önermek için teklif oluşturabilir.	Teklif başlığı, açıklaması, önerilen işlem	Teklif kaydı oluşturulur	Belirli miktarda token sahibi olmalı (örneğin minimum %1 pay)	Kullanıcı "Teklif Ver" butonuna basınca
2	oylamaBaslat	Teklif yeterli destek toplarsa oylama süreci başlatılır.	Teklif kimliği, oylama süresi	Oylama kaydı oluşturulur	Ön koşul: teklif destek limiti aşmış olmalı	Teklif desteği yeterliyse otomatik

3	oyKullan	Token sahipleri oylamada evet/hayır oyu kullanır.	Teklif kimliği, oy tercihi (evet/hayır)	Oy kaydı oluşturulur	Sadece token sahibi kullanıcılar oy kullanabilir	Oylama süresi içinde çağrılır
4	oylamaSonuclandir	Oylama süresi dolunca sonuç değerlendirilir, teklif kabul edilir veya reddedilir.	Teklif kimliği	Kabul/red sonucu kaydedilir	Oylama süresi tamamlanmış olmalı	Otomatik zaman tetiklemesi
5	onaylananTeklifiUygula	Oylamada kabul edilen teklifler, yönetim tarafından uygulanır (örneğin kâr dağıtım oranı değişimi gibi).	Teklif kimliği, işlem parametreleri	Sistem parametreleri güncellenir	Yalnızca onaylanmış teklifler uygulanabilir	oylamaSonuclandir sonrası
6	yonetimYetkisiDenetle	Sistemdeki belirli işlemler için DAO'nun yetkili olup olmadığı kontrol edilir. (Örn: Satış, yeni yatırım kararları)	İşlem kimliği	İzin/ret sonucu	İşlem öncesinde yetkilendirme gerektirir	Her kritik işlemde önce otomatik tetiklenir

7.5. Akıllı Sözleşmeler ve İş Akışları

Akıllı sözleşmeler, blok zinciri üzerinde otomatik olarak çalışan, koşullar gerçekleştiğinde kendi kendine uygulanan dijital sözleşmelerdir. Araç kiralama projesinde akıllı sözleşmeler, araç tokenizasyonundan kiralama işlemlerine, depozito yönetiminden kâr dağıtımına kadar tüm finansal ve operasyonel süreçlerde kullanılacaktır. Akıllı sözleşmeler Avalanche ağında Solidity dili ile geliştirilecek ve ERC-20 ile ERC-721 standartlarını destekleyeceklerdir.

Şirketin 3 temel iş süreci olacaktır. Birinci süreç kiralanacak araçların satın alınması, ikinci süreç satın alınan araçların kiralanması ve üçüncü olarak 3 yılını doldurmuş araçların satılması.

Şirket araba satış firmaları ya da bayileri ile sözleşmeler imzalar. Bu sözleşme ile araba satışı gerçekleştirmek isteyen firmalar 15 gün sabit fiyat garantili tekliflerini ayrıntılı olarak araba kiralama şirketinin API'lerine iletirler. Böylelikle araba satan firmaların kampanya ve

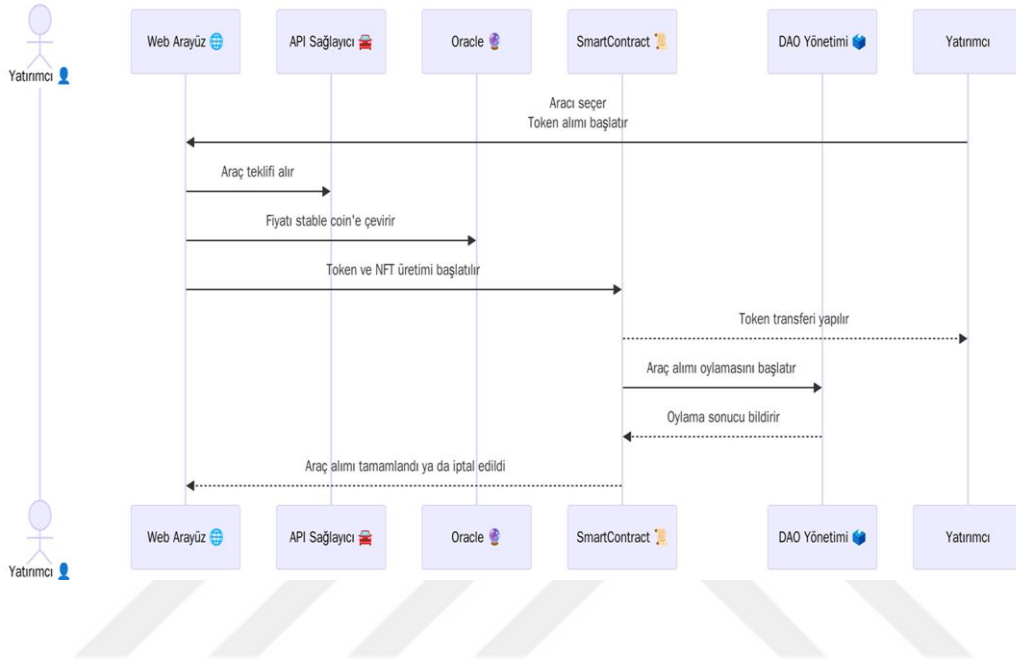
indirimlerini geniş bir kullanıcı kitlesine hızlıca sunmaları sağlanır ve veriler farklı firmalardan standart bir biçimde alınır. Şirketin araç satın alma sayfasında, araç satan firmaların apiler aracılığı ile gönderdiği teklifler işlenerek yayımlanır. Örneğin X firması tarafından A model bir araba tüm teknik özelliklerini de içeren bilgi ve örneğin 5.000.000 TL fiyat bilgisi ile şirkete gönderilir. Bu teklif sırasında sistem otomatik olarak fiyat bilgisini ve diğer bazı bilgileri istatistiki modellerle kontrol ederek yanlış veri girilmesinin önüne geçer. Bu kontrollerden geçen veriler sonrasında araba kiralama şirketinin sayfasında yayımlanır. Burada araba görselleri ve modelin teknik detayları dışında aracın fiyatı (burada sigorta, kasko, vergi ve diğer resmi giderler dahil) Avalanche üzerinden Oracle ile stable coine dönüştürülür. Örneğimizde 5.000.000 TL olan araç fiyatı vergilerle, kasko, trafik sigortası ve diğer resmi giderlerle 5.200.000 TL olsun. Oracle üzerinden alınan güncel fiyatlarla örneğin 1 USDT = 32 TL olsun, $5.200.000 \text{ TL} / 32 = 162.500 \text{ USDT}$ aracın fiyatı olacaktır. Her araç için 1000 Token üretileceği için $16,25 \text{ USDT} + 0,75 \text{ USDT}$ (işlem masrafı, bu tutar Avalanche işlem ücretleri kapsamında hesaplanacaktır.) ile araca ait bir token alınabilecektir.

Sayfada aracın resmi özellikleri ve fiyatı, projenin kapanmasına kalan süre ile birlikte kaç kişinin bu araç alımına katıldığı bilgisi de yer alır. Örneğin 200/1000 Proje tamamlanma oranı %20 gibi. Projenin bitiminde katılım yüzde seksen veya üzeri olursa araba kiralama şirketi geriye kalan tokenleri alarak araç alımı sürecinin olumlu olarak sonuçlanmasını sağlar. Araç alımı süreci kampanya bitiminde olumsuz sonuçlanırsa her katılımcıya 16,25 USDT si geri ödenir. Ancak 0,75 USDT işlem masrafları geri ödeme işlemlerinde kullanılacağı için geri ödenmez. Araba alım sürecinin olumlu sonuçlanması sonrası bu araba alım işlemi şirket yönetimin onayına sunulur. 24 saat içinde oylamaya katılan yönetim tokeni sahiplerinin salt çoğunluğunun kararına göre araç alımı gerçekleştirilir ya da katılımcılara paraları geri ödenerek süreç sona erdirilir. Bu süreçte oylamaya kimse katılmazsa araç alımı otomatik olarak olumlu sonuçlandırılır.

Araç alımına karar verilirse, yasal süreç tamamlanarak aracın alım işlemleri tamamlanır ve plakasının alındığı bilgisi sisteme manuel olarak girilir. Araç plakasının oluşması sonrası akıllı sözleşme aracın verileri saklayacak NFT ve katılımcılara dağıtılacak bu araca özel 1000 token oluşturup kullanıcıların hesabına gönderir. Böylece araç alım süreci

tamamlanmış olur. Araba alım süreci otomatik olarak sürekli devam eden bir süreçtir. Araba kiralama şirketi internet sayfasının araç alım kısmında onlarca araba sürekli yatırım için açık tutulur.

Şekil 29: Araç satın alma süreci



Araç satın alma sürecinde oluşturulacak akıllı sözleşme için aşağıdaki tabloda yer alan yol haritasının kullanılması faydalı olacaktır.

Tablo 14: Araç satın alma süreci kodlama yol haritası

	Fonksiyon / İşlem	Açıklama	Girdi Parametreleri	Çıktı Değerleri	Güvenlik Şartları	Bağılantılı İşlem / Trigger
1	fonlamaBaslat	Araç için yatırım turunu başlatır. Araç bilgileri, fiyat ve süresi belirlenir.	Araç kimliği, fiyat, fonlama süresi	Fonlama kaydı oluşturulur	Sadece yönetici çağırabilir	Yönetici başlatır
2	fonlamaKatilim	Yatırımcı stablecoin göndererek fonlamaya katılır.	Yatırım miktarı	Yatırımcı bakiyesi ve toplam fonlama güncellenir	Katılım süresi içinde olmalı, minimum yatırım sınırı olmalı	Yatırımcı "Katıl" butonuna basınca
3	fonlamaSuresiKontrol	Fonlama süresinin dolup dolmadığını ve hedefe ulaşıp	Fonlama kimliği	Başarı/başarısızlık durumu döner	Sadece süresi dolmuş fonlamalarda çalışır	Otomatik zaman tetiklemesi

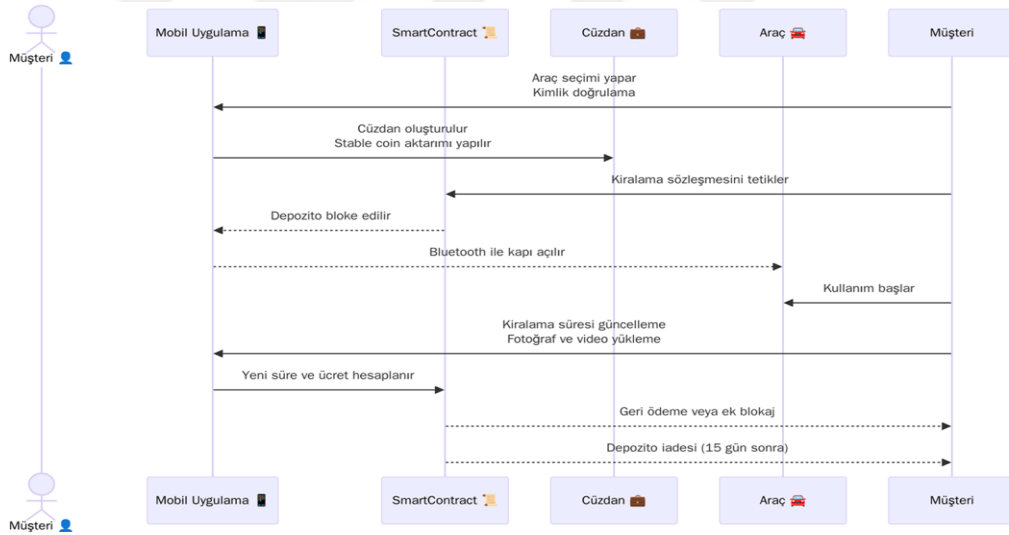
		ulaşılmadığını kontrol eder.				
4	fonlamaSonlandir	Fonlama başarılıysa şirket hesabına para aktarılır. Başarısızsa iade edilir.	Fonlama kimliği	Fonlama başarı durumu	Yalnızca yetkili kişi çalıştırabilir veya otomatik zaman tetiklemesi	Yönetici veya otomatik çağırır
5	aracTokenlariDagit	Katılımcılara yatırımlarına orantılı araç tokenları mint edilir ve cüzdanlarına gönderilir.	Yatırımcı listesi, yatırım tutarları	Token transfer işlemleri	Sadece başarılı fonlama sonrası çalıştırılabilir	fonlamaSonlandir sonrası
6	fonlamaBasarisizIsle	Fonlama başarısızsa yatırımcılara paraları iade eder.	Fonlama kimliği	İade işlemleri	Fonlama başarısız olmalı	fonlamaSuresiKontrol sonrası
7	şirketEksikPayiAl	Fonlama %80 üstündeyse şirket kalan tokenları alır ve araç alımını tamamlar.	Fonlama kimliği, eksik bakiye	Şirket için token mint edilir	%80 fonlama doluluğu sağlanmış olmalı	Otomatik veya yönetici tetikler
8	aracKaydiOlustur	Satın alınan araç blockchain üzerinde kayıt altına alınır.	Araç kimliği, araç bilgileri	Araç verisi kaydedilir	Sadece fonlama başarıyla sonlandıysa yapılır	şirketEksikPayiAl sonrası
9	kiralamaIcinHazirla	Araç kiralama sistemine aktarılır ve aktif listeye eklenir.	Araç kimliği	Kiralama listesine ekleme işlemi	Araç satın alınmış ve kaydedilmiş olmalı	aracKaydiOlustur sonrası

Araba kiralama şirketinin temel fonksiyonu olan araç kiralama, internet sayfasından ya da mobil uygulamadan müşterinin tercihi göre araç seçmesini sağlayan bir arayüz ile sunulmaktadır. Alımı tanımlanan araçlar ile müşteri kullanımında olmayan araçlar bu sayfalarda teknik özellikleri ve sistem tarafından otomatik hesaplanan kiralama fiyatları ile yayımlanır. Araç kiralama fiyatları için hesaplamalar otomatik olarak akıllı sözleşmelerle yapıldıktan sonra yönetim onayına gönderilir. Araç alımında kullanılan onay süreci burada da uygulanır.

Araç kapıları bluetooth sistemi ile açılacağı için araba kiralama kararı veren müşteriler mobil uygulama indirmek zorundadır. Mobil uygulama ile müşterinin kimlik bilgileri alınır ve müşteriye bir cüzdan tanımlanır. Müşteri bu cüzdan ile bir akıllı sözleşmeyi tetikler. Müşteri

trafik cezası, hasar ve diğer kullanıcı kaynaklı sorunlar için depozito ödemek zorundadır. Araç kiralamak isteyen müşteri mobil uygulamadan kendine yakın ve tercihinine uygun bir araç seçer. Bu aracın resimleri ve videosunu çektikten sonra tahmini araç kullanım saatini girer ve buna göre süreç fiyatlandırılır. Depozito ve tahmini fiyat bloke edilir. Müşteri arabayı cep telefonu üzerinden bluetooth teknolojisi ile açar ve kullanmaya başlar. İşinin erken bittiğini belirtirse araç kilitlenir. Müşterinin yeniden aracın resimlerini ya da videosunu çekmesi istenir. Kullanım dışı ücret müşteriye iade edilir. Müşteri aracı tahmininden uzun kullanmak isterse kendisine hatırlatma yapılır ve yeni süre kadar daha hesabından stable coin blokelenir. Depozito, 15 gün içinde herhangi bir sorun (trafik cezası, park cezası vb.) çıkmaması halinde müşteriye iletilir. Sorunlu durumlarda depozitonun sorunu gidermeye yetmemesi halinde yasal süreç başlatılır.

Şekil 30: Araç kiralama süreci



Araç kiralama sürecinde oluşturulacak akıllı sözleşme için aşağıdaki tabloda yer alan yol haritasının kullanılması faydalı olacaktır.

Tablo 15: Araç kiralama süreci kodlama yol haritası

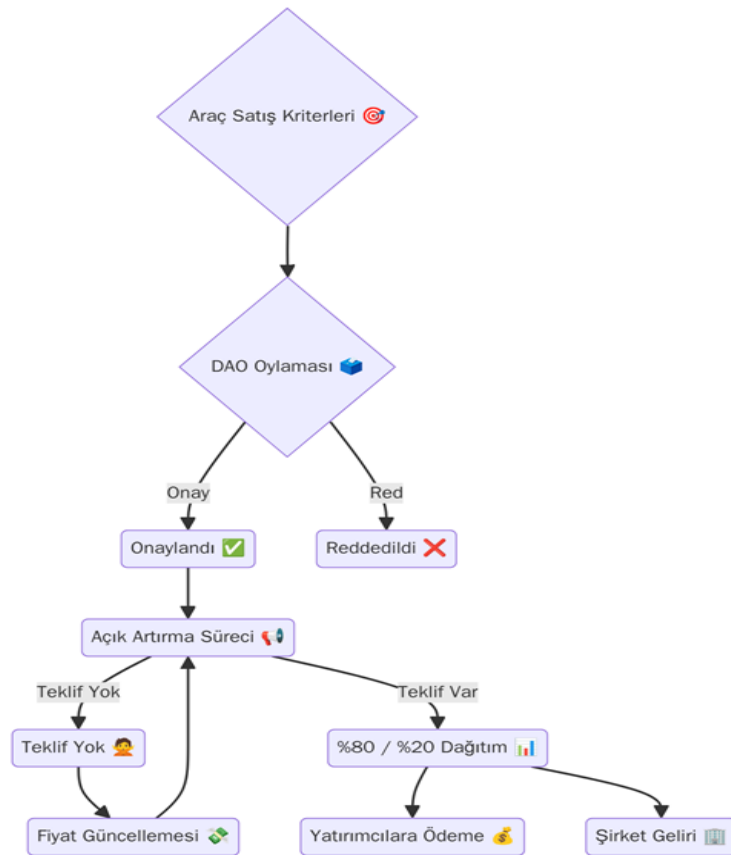
	Fonksiyon / İşlem	Açıklama	Girdi Parametreleri	Çıktı Değerleri	Güvenlik Şartları	Bağlantılı İşlem / Trigger
1	kiralamaTalepEt	Kullanıcı bir aracı kiralamak için talepte bulunur.	Araç kimliği, kiralama süresi, depozito miktarı	Kiralama talebi oluşturulur	Araç müsait olmalı, kullanıcı yeterli depozitoya sahip olmalı	Kullanıcı “Kırala” butonuna basınca
2	kiralamaOdemeOnayla	Kullanıcı kiralama bedelini ve depozitoyu akıllı sözleşmeye gönderir.	Kiralama bedeli, depozito miktarı	Fonlar akıllı sözleşmeye aktarılır	Fonlar doğru miktarda olmalı, zaman aşımı olmamalı	Talep sonrası ödeme ekranı tetiklenir
3	kiralamaBaslat	Araç kiralama süreci resmi olarak başlar, araç rezerve edilir.	Araç kimliği, kullanıcı adresi	Araç durumu “kırlandı” olur	Ödeme başarıyla alınmış olmalı	Ödeme onayından sonra otomatik tetiklenir
4	kiralamaSureTakibi	Kiralama süresi boyunca kullanım süresi izlenir, kira süresi sonunda işlemler başlatılır.	Kiralama kimliği, bitiş zamanı	Zamanlayıcı kurulumu yapılır	Başlangıç zamanı kayıtlı olmalı	kiralamaBaslat sonrası otomatik başlar
5	kiralamaBitirTalebi	Kullanıcı, kiralama süresi sonunda aracı teslim ettiğini bildirir.	Araç kimliği	Teslim talebi oluşturulur	Kiralama süresi dolmuş olmalı	Kullanıcı kiralama panelinden çağırır
6	kiralamaTeslimOnayla	Araç iade edildikten sonra şirket yetkilisi ya da akıllı sözleşme araç durumunu kontrol eder ve onaylar.	Araç kimliği, hasar durumu bilgisi	Depozito iadesi yapılır veya kesinti yapılır	Araçta hasar kontrolü yapılmalı	Yetkili veya otomatik hasar sensör verisi sonrası
7	depozitoIadeEt	Depozito (kesintili veya tam) kullanıcının cüzdanına geri gönderilir.	Kullanıcı adresi, iade miktarı	Depozito transfer işlemi	Kiralama başarılı ve araç hasarsız teslim edilmiş olmalı	kiralamaTeslimOnayla sonrası tetiklenir
8	kiralamaKaydiTamamla	Kiralama kaydı kapatılır, araç yeniden kiralanabilir duruma getirilir.	Araç kimliği	Araç durumu “müsait” yapılır	Depozito işlemi tamamlanmış olmalı	depozitoIadeEt sonrası tetiklenir

Araba kiralama şirketinin üçüncü fonksiyonu ise 3 yaşını dolduran, önemli bir kazaya karışan ya da sürekli teknik sorunlar çıkaran arabaların satılacak olmasıdır. Böylelikle müşterilere sürekli yeni ve güncel arabalarla hizmet sağlanabilecektir. Araçların 3 yaşını dolduranları, istatistiki yöntemlerle belirlenen sayıdan çok sayıda arıza verenleri ve tamir masrafları belirli tutarın üstüne çıkanları akıllı sözleşmelerle otomatik olarak satış sürecine

alınacaktır. Bu kriterlere uyan araçlar için otomatik olarak araç fiyatları belirlenecek ve Yönetim Tokeni sahiplerinin onayına sunulacaktır. Onayın kabul edilmesi durumunda araçlar açık artırma yoluyla satışa sunulacaktır. Satışı reddedilen araçlar 6 ay sonra tekrar satış için Yönetim Tokeni sahiplerinin onayına sunulacaktır.

İnternet sayfası üzerinden araç hakkında bütün bilgiler paylaşılarak başlatılan açık artırma 1 hafta sürecektir. En yüksek fiyat veren kişinin teklifi stable coin olarak bloke edilecek daha yüksek teklif gelmesi halinde yeni teklif bloke edilirken eski teklif için bloke edilen tutar serbest bırakılacaktır. Araç açık artırma süresi sonrasında otomatik olarak en yüksek teklifi veren müşteriye satılacaktır. Satış tutarının % 20'si şirket hesaplarına % 80'i ise aracın token'lerini elinde bulunduran yatırımcılara dağıtılacaktır. Hiç açık artırma teklifi almayan araçlar fiyat güncellemesi yapılarak yeniden satışa çıkarılacaktır. Bütün bu süreçler akıllı sözleşmelerle yürütülecektir.

Şekil 31: Araç satış süreci



Araç satış sürecinde oluşturulacak akıllı sözleşme için aşağıdaki tabloda yer alan yol haritasının kullanılması faydalı olacaktır.

Tablo 16: Araç satış süreci kodlama yol haritası

	Fonksiyon / İşlem	Açıklama	Girdi Parametreleri	Çıktı Değerleri	Güvenlik Şartları	Bağlantılı İşlem / Trigger
1	satisTuruBaslat	Belirli bir kriteri karşılayan araç için açık artırma usulü satış turu başlatılır.	Araç kimliği, başlangıç fiyatı, açık artırma süresi	Satış turu kaydı oluşturulur	Araç kriterlere uygun olmalı (örneğin 3 yaşını doldurmuş olmalı)	Yönetici satış başlatır
2	teklifVer	Kullanıcılar araç için teklif verir.	Teklif tutarı	Teklif kaydı oluşturulur, eski teklif serbest bırakılır	Teklif en az bir önceki tekliften yüksek olmalı	Kullanıcı “Teklif Ver” butonuna basınca
3	teklifParaTransfer	Teklifle birlikte teklif tutarı akıllı sözleşmeye gönderilir.	Teklif tutarı	Fonlar akıllı sözleşmede bloke edilir	Doğru teklif miktarı olmalı	Teklif verildiğinde otomatik tetiklenir
4	acikArtirmaSuresiKontrol	Açık artırmanın süresi dolunca en yüksek teklif belirlenir.	Satış turu kimliği	En yüksek teklif kaydı	Süre dolmuş olmalı	Otomatik zaman tetiklemesi
5	tokenSahipleriOylamaBaslat	Satış sonrası token sahipleri araç satışına onay verip vermemek için oylamaya çağrılır.	Araç kimliği	Oylama başlatılır	Satış süresi dolmuş ve teklif alınmış olmalı	acikArtirmaSuresiKontrol sonrası
6	oylamaSonuclandir	Oylama sonucu değerlendirilir: çoğunluk onayı varsa satış tamamlanır.	Araç kimliği	Satış onayı veya iptali belirlenir	Belirli çoğunluk eşiği (örneğin %50+) aşılmalı	Oylama süresi sonunda
7	satisTamamla	Oylama onaylıysa aracı satın alan kişiye transfer yapılır, token sahiplerine gelir dağıtılır.	Araç kimliği, alıcı adresi	Satış işlemi tamamlanır, gelir dağıtılır	Oylama onayı alınmış olmalı	oylamaSonuclandir sonrası

8	tokenSahiplerineOde meYap	Satıştan elde edilen gelir token sahiplerine oranlı şekilde dağıtılır. Şirket %10 komisyon alır.	Token sahipleri listesi	Gelir transfer işlemleri	Satış başarıyla tamamlanmış olmalı	satisTamamla sonrası
9	aracKaydiKapat	Satış tamamlanan aracın kaydı kapatılır, sistemden çıkarılır.	Araç kimliği	Araç kaydı silinir veya arşivlenir	Satış başarıyla tamamlanmış olmalı	tokenSahipler ineOdemeYa p sonrası

Projemizde 5 farklı akıllı sözleşme için kodlama yol haritaları verildi ve 38 farklı işlem/fonksiyon tanımlandı. Bu akıllı sözleşmelerin Solidity diliyle yazılacağını öngördük. Çünkü bu dil hem Ethereum hem de Avalanche üzerinde çalışabilen bir akıllı sözleşme kodlama dilidir. Solidity komutları Ethereum’da Ethereum Virtual Machine (EVM) ile çalıştırılırken Avalanche’da (C-Chain) Ethereum uyumlu C-Chain EVM ile çalıştırılır. Bu kapsam da işlemler için hesaplanan “Tahmini Gas Tüketimi” iki zincirde de aynı hesaplanmaktadır. Burada farkı yaratan Ethereum gas ücreti ile Avalanche gas ücretidir. Tezin genel olarak yazıldığı 2024 yılında Ethereum da ortalama gas fiyatları yaklaşık 30 Gwei (Etherscan Gas Tracker) olmuştur. Avalanche (C-Chain) için ise bu rakam 2 Gwei (Avalanche Explorer) olmuştur. Bu bakış açısıyla yaklaşık 15 kat daha ucuz olduğu için akıllı sözleşmelerin Avalanche ağında hazırlanması öngörülmüştür. Ancak Nisan 2025 rakamlarına bakıldığında Ethereum’da gas ücretlerinin 2 Gwei’nin altına düştüğü görülmektedir. Burada sözleşmelerin solidity ile yazılmış olması sayesinde işlemler daha ucuz olan ağda gerçekleştirilebilir. Şimdi sözleşmelerde tahmini gas tüketimi hesaplamalarını inceleyelim.

Tablo 17: Akıllı sözleşme fonksiyonları/işlemleri için tahmini gas tüketimi

	Fonksiyon / İşlem	Tahmini Gas Tüketimi	Ethereum Maliyeti (30 Gwei)	Avalanche Maliyeti (2 Gwei)
1	fonlamaBaslat	100.450	0.0000030135 ETH	0.000000201 AVAX
2	fonlamaKatilim	31.400	0.000000942 ETH	0.0000000628 AVAX
3	fonlamaSuresiKontrol	2.100–7.100	0.000000063–0.000000213 ETH	0.0000000042–0.0000000142
4	fonlamaSonlandir	28.500	0.000000855 ETH	0.000000057 AVAX
5	aracTokenlariDagit	26.400 × yatırımcı	0.000000792 × yatırımcı ETH	0.0000000528 × yatırımcı
6	fonlamaBasarisizIsle	26.400 × yatırımcı	0.000000792 × yatırımcı ETH	0.0000000528 × yatırımcı
7	sirketEksikPayiAl	26.400	0.000000792 ETH	0.0000000528 AVAX
8	aracKaydiOlustur	100.400	0.000003012 ETH	0.0000002008 AVAX

9	kiralamaIcinHazirla	5.400	0.000000162 ETH	0.0000000108 AVAX
10	kiralamaTalepEt	62.500	0.000001875 ETH	0.000000125 AVAX
11	kiralamaOdemeOnayla	26.400	0.000000792 ETH	0.0000000528 AVAX
12	kiralamaBaslat	10.400	0.000000312 ETH	0.0000000208 AVAX
13	kiralamaSureTakibi	4.200	0.000000126 ETH	0.0000000084 AVAX
14	kiralamaBitirTalebi	5.400	0.000000162 ETH	0.0000000108 AVAX
15	kiralamaTeslimOnayla	10.400	0.000000312 ETH	0.0000000208 AVAX
16	depozitoIadeEt	26.400	0.000000792 ETH	0.0000000528 AVAX
17	kiralamaKaydiTamamla	10.400	0.000000312 ETH	0.0000000208 AVAX
18	satisTuruBaslat	80.400	0.000002412 ETH	0.0000001608 AVAX
19	teklifVer	28.500	0.000000855 ETH	0.000000057 AVAX
20	teklifParaTransfer	26.400	0.000000792 ETH	0.0000000528 AVAX
21	acikArtirmaSuresiKontrol	2.100	0.000000063 ETH	0.0000000042 AVAX
22	tokenSahipleriOylamaBaslat	60.400	0.000001812 ETH	0.0000001208 AVAX
23	oylamaSonuclandir	11.400	0.000000342 ETH	0.0000000228 AVAX
24	satisTamamla	10.400	0.000000312 ETH	0.0000000208 AVAX
25	tokenSahiplerineOdemeYap	$26.400 \times \text{token sahibi}$	$0.000000792 \times \text{token sahibi}$	$0.0000000528 \times \text{token sahibi}$
26	aracKaydiKapat	17.000–20.000	0.00000051–0.0000006 ETH	0.000000034–0.00000004
27	aracBilgisiKaydet	80.000–100.000	0.0000024–0.000003 ETH	0.00000016–0.0000002 AVAX
28	tokenlestirmeBaslat	40.400	0.000001212 ETH	0.0000000808 AVAX
29	tokenSembolOlustur	40.400	0.000001212 ETH	0.0000000808 AVAX
30	tokenlariMintEt	20.400	0.000000612 ETH	0.0000000408 AVAX
31	tokenlariKitle	5.400	0.000000162 ETH	0.0000000108 AVAX
32	yatirimTuruneHazirla	5.400	0.000000162 ETH	0.0000000108 AVAX
33	yonetimOnerisiOlustur	60.400	0.000001812 ETH	0.0000001208 AVAX
34	oylamaBaslat	40.400	0.000001212 ETH	0.0000000808 AVAX
35	oyKullan	10.400	0.000000312 ETH	0.0000000208 AVAX
36	onaylananTeklifiUygula	5.400–10.400	0.000000162–0.000000312 ETH	0.0000000108–0.0000000208
37	yonetimYetkisiDenetle	2.100	0.000000063 ETH	0.0000000042 AVAX
38	notifyVehicleReadyForRent	5.400	0.000000162 ETH	0.0000000108 AVAX

Burada fonksiyonların gas tüketimi “4.2. Ethereum ve Akıllı Kontratlar” başlığında verilen Ethereum yellowpaper gas hesaplama değerleri kullanılarak hesaplanmaktadır. Örneğin “fonlamaBaslat” fonksiyonu aşağıdaki gibi hesaplanmıştır.

Tablo 18: fonlamaBaslat fonksiyonu gas tüketimi hesaplaması

Fonksiyon / Alt İşlem	Gas Tüketimi	Ethereum Maliyeti (30 Gwei)	Avalanche Maliyeti (2 Gwei)
fonlamaBaslat (Toplam)	100.450 gas	0.0000030135 ETH	0.000000201 AVAX
SSTORE (araç ID kaydı)	20.000 gas	0.0000006 ETH	0.00000004 AVAX
SSTORE (fiyat kaydı)	20.000 gas	0.0000006 ETH	0.00000004 AVAX
SSTORE (fonlama süresi kaydı)	20.000 gas	0.0000006 ETH	0.00000004 AVAX
SSTORE (hedef yatırım miktarı kaydı)	20.000 gas	0.0000006 ETH	0.00000004 AVAX
SSTORE (aktiflik durumu kaydı)	20.000 gas	0.0000006 ETH	0.00000004 AVAX
LOG1 (Fonlama başlatıldı event emit)	400 gas	0.000000012 ETH	0.000000008 AVAX
Basit matematik ve kontrol işlemleri (require, karşılaştırmalar)	50 gas	0.0000000015 ETH	0.0000000001 AVAX

Bir başka örnek olarak da “fonlamaKatilim” fonksiyonu aşağıdaki tabloda incelenmektedir.

Tablo 19: fonlamaKatilim fonksiyonu gas tüketimi hesaplaması

Fonksiyon / Alt İşlem	Gas Tüketimi	Ethereum Maliyeti (30 Gwei)	Avalanche Maliyeti (2 Gwei)
fonlamaKatilim (Toplam)	31.400 gas	0.000000942 ETH	0.0000000628 AVAX
CALL (ETH/ERC20 transfer işlemi)	21.000 gas	0.00000063 ETH	0.000000042 AVAX
SSTORE (yatırımcı yatırımı kaydı update)	5.000 gas	0.00000015 ETH	0.00000001 AVAX
SSTORE (toplam fonlama miktarı update)	5.000 gas	0.00000015 ETH	0.00000001 AVAX
LOG1 (Yeni katılım event emit)	400 gas	0.000000012 ETH	0.000000008 AVAX

Günümüz teknolojisinde ve blok zinciri teknolojisinin geldiği nokta açısından bu proje oldukça uygulanabilir bir projedir. Burada araba satan firmaların yasal işlemleri halletmesi tamir ve bakım işlemlerinin servislerle anlaşarak arabaların bulunduğu lokasyonlarda

yapılması ya da servis firmaları tarafından araçların toplanarak bakımı yapılması gibi iş modelleriyle neredeyse hiç insan müdahalesi olmayan bir sistem kurgulanabilir.

Gerçek hayat uygulamalarında bu işlemlerin yapılabilmesi için yasal mercilerden izin alınması gerekmektedir. Özellikle araçların tokenleştirilmesi konusu SPK'nın uhdesinde olan bir konu olduğu gibi işlemlerin USDT üzerinden yapılması ve işin vergi boyutu gibi konular oldukça dikkatli incelenmesi ve mevzuata uygun şekilde kurgulanması gereken konulardır.

Bu hususlar dışında örneğimiz blok zinciri teknolojisinin nasıl hayatımızı şekillendirebileceğine dair oldukça tutarlı bir tasarımdır. Zaman içerisinde birçok firmanın hizmet tarzının ve mevzuatsal düzenlemelerin bu tip iş modellerine izin verecek şekilde olgunlaşacağı aşikardır.

Sonuç olarak blok zinciri teknolojisi ile tamamen kuralları akıllı sözleşmeler tarafından belirlenmiş ve kimsenin bu kuralları değiştiremeyeceği veya kendi menfaatine bir uygulamaya gidemeyeceği bir yapı kurgulanmıştır. Bütün süreç ve işlemlerin blok zincirine kaydedildiği ve bu kayıtların herkes tarafından takip edilebildiği bir mimari söz konusudur. Hem yatırım yapmak isteyen hem de araba kiralamak isteyenler için bütün verilerin açık olduğu merkeziyetsiz bir organizasyon oluşturulmuştur.

8. SONUÇ VE DEĞERLENDİRMELER

Bu tez çalışması, blok zinciri teknolojisinin yalnızca yüzeysel uygulamalarını değil, aynı zamanda onun temelini oluşturan matematiksel yapıları ve kavramsal temellerini ortaya koymayı amaçlamıştır. Dijitalleşmenin ivme kazandığı bir çağda, veri güvenliğinden finansal sistemlere, oylama süreçlerinden sanat eserlerine kadar geniş bir etki alanına sahip olan blok zinciri, artık yalnızca teknolojik bir yenilik değil, toplumsal bir dönüşümün aracı hâline gelmektedir.

Bu bağlamda, tezde ilk olarak teknolojinin tarihsel kökeni ve gelişim seyri ele alınmaktadır. Blok zincirinin Bitcoin ile başlayan popülerleşme süreci, yalnızca bir kripto para teknolojisinden öteye geçerek, merkeziyetsiz organizasyon yapılarının, şeffaf yönetim modellerinin ve itibari güvene dayalı olmayan sistemlerin doğmasına neden olmaktadır. Ancak bu dönüşüm, yalnızca teknik bir ilerleme değildir; aynı zamanda ekonomik, sosyal ve politik yönleri olan bir paradigma değişimi olarak karşımıza çıkmaktadır.

Blok zinciri teknolojisinin güvenlik, doğruluk ve bütünlük gibi temel fonksiyonlarının sürdürülebilirliğinin, doğrudan matematiksel yapılarla ilişkili olduğu görülmektedir. Bu tezde ele alınan temel matematiksel araçlardan biri olan kriptografi, sistemin neredeyse tüm işleyişini belirleyen birincil disiplin olarak değerlendirilebilir. Özellikle asimetrik anahtar kriptografisi, dijital imzalar ve özetleme fonksiyonları (hashing) gibi kavramlar, verilerin sadece yetkili kişilerce doğrulanmasını sağlamak ve işlemlerin geri döndürülemezliğini garanti altına almaktadır. Örneğin SHA-256 gibi özetleme algoritmaları, sabit uzunluklu ve tek yönlü veri üretimi sayesinde blok zincirinin tutarlılığını sağlamaktadırlar. Bu algoritmaların arkasındaki matematiksel temeller (örneğin çakışma (collision) olasılığı, deterministiklik, rastgelelik düzeyi) sistemin saldırılara karşı ne derece dirençli olduğunu belirlemektedir. Merkle ağaçları gibi veri yapıları ise çok büyük veri setlerinin blok zincirine verimli ve güvenli biçimde dahil edilmesini mümkün kılmakta ve verilerin doğruluğu daha az işlem yapılarak ispatlanabilmektedir. Bütün bu bileşenler, yalnızca pratik avantajlar değil, aynı zamanda sistemin formal güvenlik ispatlarının yapılabilmesine olanak tanıyan matematiksel argümanlardır.

Tezde sunulan matematiksel içerik, her ne kadar sistemin genel mantığını ortaya koymak için yeterli olsa da, daha ileri düzey araştırmalar için derinleştirilmeye açık pek çok alan barındırmaktadır. Örneğin dijital imza algoritmalarının formel güvenlik modelleri, karmaşıklık sınıfları çerçevesinde değerlendirilerek analiz edilebilir. Eliptik Eğri Kriptografisi (ECC) gibi modern şifreleme teknikleri, yalnızca hızlı ve güvenli oldukları için değil, aynı zamanda küçük anahtar boyutlarında bile yüksek güvenlik seviyesi sundukları için tercih edilmektedirler. Bu yapılar cebirsel eğriler, modüler aritmetik ve grup kuramı üzerine kuruludur ve bu konuların daha derin matematiksel açıklamalar eşliğinde incelenmesi, konunun matematiksel derinliğinin kavranmasını sağlayacaktır. Ayrıca sıfır bilgi ispatları (Zero Knowledge Proofs), homomorfik şifreleme ve çok taraflı hesaplama gibi gelişmiş protokoller, özellikle blok zincirinin gizlilik, ölçeklenebilirlik ve işlem doğruluğu gibi alanlarında matematiksel çözüm modelleri sunmaktadır. Bu yöntemlerin blok zinciri ile entegrasyonu, sadece uygulama düzeyinde değil, aynı zamanda teorik düzlemde de geleceğe dönük özgün katkılar barındırmaktadır. Bu bağlamda tez, mevcut hâliyle sağlam bir temel sunsa da, matematiksel ispat teknikleriyle desteklenmiş ek analizlerin ilerideki akademik çalışmalar için değerli bir kaynak hâline gelebileceği düşünülmektedir.

Blok zinciri sistemlerinin toplumsal hayata entegrasyonunda karşılaşılan başlıca sorunlar çalışmanın önemli bir bölümünü oluşturmaktadır. Kripto paraların güncellenmesi, yazılım hatalarının düzeltilmesi veya yeni protokollerin devreye alınması gibi teknik değişimlerde bu kararları kimin aldığı, hangi yöntemle aldığı ve hangi meşruiyete dayandığı büyük bir tartışma konusu olmaktadır. Merkeziyetsizliği ilke edinmiş bir yapı içerisinde, güncellemelerin belirli yazılım gruplarının veya büyük madencilerin elinde olması, hem teknik hem etik sorunlara yol açmaktadır. Bu durum, sözde dağıtık olan sistemlerin fiiliyatta nasıl bir güç yoğunlaşmasına maruz kaldığını göstermektedir.

Tezde ele alınan NFT, DAO, DeFi, merkeziyetsiz kimlik sistemleri, tedarik zinciri izleme, sağlık verilerinin korunması gibi blok zinciri uygulamaları, her ne kadar teknolojik yenilikler sunuyor olsa da, kripto paralar kadar geniş kitlelere ulaşamamıştır. Bu tür uygulamaların büyük çoğunluğu hâlâ belirli topluluklarla sınırlı, yerel projeler düzeyinde kalmakta; global ölçekte etkili olamamaktadır. Bu durumun altında yatan temel paradoks, blok zinciri

teknolojisinin sistemin dışından gelip mevcut otoritelere alternatif bir yapı sunmaya çalışmasıdır. Ancak gerçek dünyada işleyen sistemler, yasal düzenlemeler, finansal ağlar ve kullanıcı alışkanlıkları belirli otoriteler tarafından yönetilmektedir. Bu nedenle otoritelerin desteğini almadan onların işlevini devralmak isteyen bir teknoloji, ironik biçimde yaygınlaşmak için yine otoritelere ihtiyaç duymaktadır. Bu çelişki, blok zinciri projelerinin önünde hem teknik hem de sosyo-politik bir engel olarak durmaktadır.

Blok zinciri sistemlerinin temel vaadi, aracıya ihtiyaç duymadan işlem yapılabilmesidir. Ancak uygulamada bu sistemler, ortalama bir kullanıcı için oldukça teknik ve anlaşılması güç yapılardır. Kriptografik anahtar yönetimi, cüzdan kurulumu, “gas” ücretlerinin hesaplanması ve akıllı sözleşmelerle etkileşim gibi konular teknik bilgi gerektirmektedir. Bu nedenle, sistemin teorik olarak "aracısız" olması, pratikte "yeni tip araçlara" kapı aralamaktadır. Kullanıcılar genellikle bu süreçlerde kripto borsaları, cüzdan sağlayıcıları veya danışman kişi ve kuruluşlara başvurmaktadır. Bu durum, sistemin merkeziyetsizliğini teknik düzeyde korurken, sosyolojik düzeyde yeniden merkezileşmeye neden olmaktadır. Böylece kullanıcılar, blok zincirinin sağladığı güveni tam anlamıyla doğrudan değil, hâlâ üçüncü kişilere dayalı bir güven mekanizması üzerinden kurmak zorunda kalmaktadır.

Bir başka önemli boyut da blok zinciri sistemlerine duyulan güvenin doğasıdır. Klasik finansal yapılarda bireyler, devletin veya merkez bankasının güvencesine dayanarak işlemlerde bulunurken, blok zinciri sistemlerinde bu güven tamamen kriptografik doğrulama, ağ yapısı ve protokol güvenliği üzerinden sağlanmaktadır. Ancak bu yapıların arkasında bir devletin veya kurumsal yapının olmaması, özellikle insan psikolojisi açısından ciddi bir belirsizlik doğurmaktadır. Kullanıcılar sistemin “matematiksel olarak güvenli” olduğunu bilseler dahi, bu güveni “hissedebilmek” konusunda sorun yaşayabilmekte; bu da kullanıcı davranışlarını doğrudan etkileyen önemli bir faktör olmaktadır. Bu yönüyle blok zinciri teknolojisinin yalnızca teknik değil, aynı zamanda sosyolojik ve psikolojik analizlere de ihtiyaç duyduğu açıktır.

Tezin uygulama kısmında önerilen “tokenizasyon temelli araç kiralama modeli”, blok zinciri altyapısının sadece para transferiyle sınırlı kalmadığını, gerçek dünyadaki somut sorunlara da çözüm üretebilecek esneklikte olduğunu göstermektedir. Bu model, yatırımcıları,

kullanıcıları ve platformu bir araya getirerek şeffaf, denetlenebilir ve güvenilir bir ekonomik sistem oluşturmayı amaçlamaktadır. Bununla birlikte, bu tür sistemlerin pratikte uygulanabilirliği, regülasyon, kullanıcı deneyimi, vergi sistemine entegrasyon ve kötü niyetli kullanımlara karşı önlemler gibi pek çok ek başlık altında derinlemesine değerlendirilmelidir.

Ayrıca, DAO (Merkeziyetsiz Otonom Organizasyon), NFT (Nitelikli Fikri Tapu), DeFi (Merkeziyetsiz Finans) gibi güncel yapılarla ilgili yapılan analizler, teknolojinin sınırsız potansiyelini ortaya koyarken, aynı zamanda bu yapıların henüz tam olgunlaşmamış ve birçok yönden denetimsiz olduğunu da göstermektedir. Özellikle DeFi sistemlerindeki yüksek getiri vaatleri, kullanıcıları manipülasyona açık hâle getirebilmekte ve “ponzi” benzeri yapılarla kolayca sömürülebilmektedir. Bu durum, sadece teknik bilgi değil, aynı zamanda etik ilkeler ve regülasyon ihtiyacını da gündeme getirmektedir.

Blok zinciri teknolojisi, yalnızca kendi başına bir veri yapısı değil, aynı zamanda gelecekteki birçok yapının temelini oluşturacak altyapı katmanı olarak değerlendirilmektedir. Metaverse evrenlerinde kullanıcı kimliklerinin doğrulanması, dijital varlık sahipliği ve sınır ötesi mikro ödemeler gibi süreçler doğrudan blok zinciri üzerinde çalışmaktadır. ETF’ler gibi geleneksel finansal araçlar da blok zinciri üzerinde tokenize edilerek şeffaf ve izlenebilir hâle getirilmektedir. Web 3.0 yaklaşımıyla birlikte kullanıcı verilerinin kontrolü tekrar bireylere geçerken, IoT cihazları ile blok zinciri entegrasyonu sayesinde cihazların güvenli veri alışverişi sağlanması hedeflenmektedir. Bu cihazlardan elde edilen verilerin blok zinciri için gerçek hayat verileri sağlayacak yeni oracle’lar olacağı da unutulmamalıdır. Tüm bu entegrasyonlar, blok zincirinin sadece bir ödeme teknolojisi değil, aynı zamanda yeni nesil internetin omurgası olma potansiyeline de sahip olduğunu göstermektedir.

Yapay zekâ teknolojilerinin son yıllarda hızlı gelişimi, kamuoyunun dikkatini blok zinciri teknolojisinden bir ölçüde uzaklaştırmış olsa da bu iki alanın kesişim kümesi giderek büyümektedir. Yapay zekâ, blok zincirinin anlaşılması zor teknik süreçlerini sadeleştirmek ve kullanıcılara erişilebilir hâle getirmek açısından büyük bir potansiyel taşımaktadır. Gelecekte her kullanıcının, kişiselleştirilmiş bir yapay zekâ aracılığıyla cüzdan yönetimi yapabilmesi, gas ücretlerini optimize edebilmesi, akıllı sözleşmeleri otomatik olarak

yazdırabilmesi ve risk analizi gerçekleştirebilmesi mümkün görülmektedir. Bu durum, teknik yeterliliği olmayan bireylerin bile merkeziyetsiz yapılarla güvenli şekilde etkileşim kurmasını sağlayacaktır. Ayrıca yapay zekânın büyük veri analizi kabiliyeti sayesinde zincir üzerindeki anomaliler, saldırı girişimleri ve dolandırıcılık faaliyetleri çok daha etkili biçimde tespit edilebilir. Yapay zekânın blok zinciri ile bu şekilde bütünleşmesi hem kullanım kolaylığı sağlayacak hem de teknolojinin sürdürülebilirliğini destekleyecektir. Dolayısıyla kısa vadede dikkatleri blok zincirinden uzaklaştırmış gibi görünse de uzun vadede bu iki alanın birleşimi, dijital dönüşümün en güçlü bileşenlerinden biri olabilecektir.

Sonuç olarak, bu çalışma blok zinciri teknolojisini matematiksel, teknik ve kavramsal olarak kapsamlı bir şekilde farklı boyutlarla ele almakta; bir yandan sistemin nasıl çalıştığını açıklarken, diğer yandan bu yapıların toplumla olan etkileşimini, potansiyel faydalarını ve sınırlılıklarını da göz önünde sunmaktadır. Blok zinciri teknolojileri, gelecekteki ekonomik ve yönetsel yapılarda köklü dönüşümlere neden olabilecek potansiyele sahiptir. Ancak bu dönüşümün sağlıklı bir şekilde gerçekleşebilmesi için, yalnızca yazılım ve donanım yeterli olmayacaktır. Aynı zamanda güvenlik, regülasyon, kullanıcı eğitimi ve etik tasarım gibi çok sayıda faktörün birlikte düşünülmesi gerekmektedir.

Bu tez, blok zincirini yalnızca bir teknoloji olarak değil, aynı zamanda disiplinler arası bir düşünce alanı olarak ele alarak, gelecekte yapılacak daha derinlikli çalışmalar için kapsamlı bir kaynak sunmayı amaçlamaktadır.

KAYNAKLAR

- [1] X. Yi, X. Yang, A. Kelarev, K.Y. Lam, Z. Tari, Blockchain Foundations and Applications, Springer, Cham, 2022.
- [2] W. Diffie, M.E. Hellman, New Directions in Cryptography, IEEE Transactions on Information Theory, IT-22 (1976) 644-654.
- [3] R.L. Rivest, A. Shamir, L. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, 21(2) (1978) 120-126.
- [4] D.L. Chaum, Computer Systems Established, Maintained and Trusted by Mutually Suspicious Groups, 1982.
- [5] D. Chaum, Security without Identification: Transaction Systems to Make Big Brother Obsolete, Communications of the ACM, 28(10) (1985) 1030-1044.
- [6] S. Haber, W.S. Stornetta, How To Time-Stamp a Digital Document, Journal of Cryptology, 3 (1991) 99-111.
- [7] C. Dwork, M. Naor, Pricing via Processing or Combatting Junk Mail, in: Annual International Cryptology Conference, 1992.
- [8] M. Jakobsson, A. Juels, Proofs Of Work And Bread Pudding Protocols, 1999.
- [9] S. Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.
- [10] B. Block, <https://www.blockchain.com/explorer/blocks/btc/170> (Eriřim tarihi: 23 Mart 2025).
- [11] Whois, <https://www.whois.com> (Eriřim tarihi: 27 Kasım 2024).
- [12] Anonim, JPG File Sells for \$69 Million, as ‘NFT Mania’ Gathers Pace, The New York Times, <https://www.nytimes.com/2021/03/11/arts/design/nft-auction-christies-beeples.html> (Eriřim tarihi: 25 Nisan 2025).
- [13] Anonim, Coinmarketcap Academy Article, <https://coinmarketcap.com/academy/tr/article/largest-crypto-heists-in-history-have-exchanges-learned-everything-from-their-mistakes> (Eriřim tarihi: 26 Mart 2025).
- [14] Anonim, DAO Attack Analysis, <https://www.deloitte.com/ie/en/Industries/technology/research/DAO-Attack-Analysis.html> (Eriřim tarihi: 26 Mart 2025).

- [15] Anonim, Axie Infinity 625 Million Dollar Hack Explained, <https://www.coinbase.com/bytes/archive/axie-infinity-625-million-dollar-hack-explained> (Erişim tarihi: 26 Mart 2025).
- [16] Anonim, Wormhole Hit with Possible \$320 Million Hack, <https://www.reuters.com/technology/crypto-network-wormhole-hit-with-possible-320-mln-hack-2022-02-03/> (Erişim tarihi: 26 Mart 2025).
- [17] Anonim, Crypto Exchange Hacks, <https://www.kaspersky.com.tr/resource-center/threats/crypto-exchange-hacks> (Erişim tarihi: 26 Mart 2025).
- [18] Anonim, Hong Kong Crypto Firm Hit by \$200 Million Hack, <https://www.reuters.com/technology/hong-kong-crypto-firm-hit-by-200-million-hack-2023-09-25/> (Erişim tarihi: 26 Mart 2025).
- [19] Anonim, Kuzey Kore 1.5 Milyar Dolarlık Kriptoyu Nasıl Çaldı, <https://tr.euronews.com/next/2025/03/02/tarihin-en-buyuk-vurgunu-kuzey-kore-15-milyar-dolarlik-kriptoyu-nasil-caldi> (Erişim tarihi: 26 Mart 2025).
- [20] W. Stallings, Cryptography and Network Security: Principles and Practice, Prentice Hall Press, United States, 2013, p. 317.
- [21] C. Paar, J. Pelzl, Understanding Cryptography: A Textbook for Students and Practitioners, Springer, Berlin–London, 2009, pp. 293, 319.
- [22] D. Boneh, V. Shoup, A Graduate Course in Applied Cryptography, 2017, p. 283.
- [23] R. Merkle, Secrecy, authentication, and public key systems, Ph.D. Thesis, Stanford University, 1979.
- [24] W. Penard, T. van Werkhoven, On the Secure Hash Algorithm Family, in: Cryptography in Context, Wiley, Hoboken NJ USA, 2008, 1–18.
- [25] NIST, Secure Hash Standard, <https://csrc.nist.gov/files/pubs/fips/180-2/final/docs/fips180-2.pdf> (Erişim tarihi: 20 Mart 2025).
- [26] TeamKeccak, <https://keccak.team/> (Erişim tarihi: 29 Mart 2025).
- [27] B. Preneel, Davies–Meyer Hash Function, in: Encyclopedia of Cryptography and Security, Springer, Boston, MA, 2005.
- [28] R.V. Mises, Über Aufteilungs- und Besetzungswahrscheinlichkeiten, Revue de la faculté des sciences de l'Université d'Istanbul, (1939) p. 2025.

- [29] M. Abramowitz, I.A. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, Dover Publications, New York, 1970, pp. 69, 4.2.1.
- [30] H. Delfs, H. Knebl, Introduction to Cryptography: Principles and Applications, Springer, 2007.
- [31] A.J. Menezes, P.C. van Oorschot, S.A. Vanstone, Public-Key Encryption, in: Handbook of Applied Cryptography, CRC Press, 1996, 425–488.
- [32] W. Diffie, M.E. Hellman, New Directions in Cryptography, IEEE Transactions on Information Theory, IT-22 (1976) 644–654.
- [33] J. Hoffstein, J. Pipher, J.H. Silverman, An Introduction to Mathematical Cryptography, Springer, 2008, pp. 16, 114.
- [34] G.L. Miller, Riemann's Hypothesis and Tests for Primality, Journal of Computer and System Sciences, 13(3) (1976) 300–331.
- [35] M.O. Rabin, Probabilistic Algorithm for Testing Primality, Journal of Number Theory, 12(1) (1980) 128–138.
- [36] N. Koblitz, Elliptic Curve Cryptosystems, Mathematics of Computation, 48(177) (1987) 203–209.
- [37] V.S. Miller, Use of Elliptic Curves in Cryptography, in: Advances in Cryptology — CRYPTO '85, LNCS 85, (1986) 417–426.
- [38] D. Hankerson, A. Menezes, Elliptic Curve Cryptography, in: Encyclopedia of Cryptography and Security, 2nd Ed., Springer, Boston, MA, 2011.
- [39] NIST, Digital Signature Standard (DSS), <https://web.archive.org/web/20131213131144/http://www.itl.nist.gov/fipspubs/fip186.htm> (Erişim tarihi: 30 Mart 2025).
- [40] A.M. van S.V.D. Johnson, The Elliptic Curve Digital Signature Algorithm (ECDSA), International Journal of Information Security, 1(1) (2001) 36–63.
- [41] NIST, Securing Web Transactions TLS Server Certificate Management, NIST SP 1800-16, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1800-16.pdf> (Erişim tarihi: 30 Mart 2025).
- [42] A. Çevik, Kuantum Hesaplama Kuramı, ODTÜ Yayıncılık, Ankara, 2021.

- [43] NIST, NIST Releases First 3 Finalized Post-Quantum Encryption Standards, <https://www.nist.gov/news-events/news/2024/08/nist-releases-first-3-finalized-post-quantum-encryption-standards> (Eriřim tarihi: 30 Mart 2025).
- [44] B. Schoenmakers, Lecture Notes Cryptographic Protocols, <https://berry.win.tue.nl/CryptographicProtocols/LectureNotes.pdf>, 30 (Eriřim tarihi: 30 Mart 2025).
- [45] Anonim, Monero, <https://www.getmonero.org/> (Eriřim tarihi: 23 řubat 2025).
- [46] B. Schoenmakers, Lecture Notes Cryptographic Protocols, <https://berry.win.tue.nl/CryptographicProtocols/LectureNotes.pdf>, 34 (Eriřim tarihi: 30 Mart 2025).
- [47] B. Schoenmakers, Lecture Notes Cryptographic Protocols, <https://berry.win.tue.nl/CryptographicProtocols/LectureNotes.pdf>, 47 (Eriřim tarihi: 30 Mart 2025).
- [48] E. Ben-Sasson et al., Zerocash: Decentralized Anonymous Payments from Bitcoin, in: IEEE Symposium on Security and Privacy (IEEE S&P), 2014, Berkeley, CA, 459–474.
- [49] E. Ben-Sasson, A. Chiesa, E. Tromer, M. Virza, Succinct Non-Interactive Zero Knowledge for a von Neumann Architecture, 2013.
- [50] E. Ben-Sasson, I. Bentov, Y. Horesh, M. Riabzev, Scalable, Transparent, and Post-Quantum Secure Computational Integrity, 2018.
- [51] Blockchain.com, 889503. Blok, <https://www.blockchain.com/explorer> (Eriřim tarihi: 26 Mart 2025).
- [52] A.M. Antonopoulos, Mastering Bitcoin, O'Reilly Media, California, 2014.
- [53] Bitcoin Wiki, <https://en.bitcoin.it/wiki/Multi-signature> (Eriřim tarihi: 30 Mart 2025).
- [54] L. Lamport, R. Shostak, M. Pease, The Byzantine Generals Problem, ACM Transactions on Programming Languages and Systems, 4(3) (1982) 382–401.
- [55] R.C. Merkle, A Digital Signature Based on a Conventional Encryption Function, in: Advances in Cryptology – CRYPTO '87, Lecture Notes in Computer Science, 1988.
- [56] T. Lange, Hash-Based Signatures, in: Encyclopedia of Cryptography and Security, Springer, 2011.
- [57] W. Zhao, S. Yang, X. Luo, J. Zhou, On PeerCoin Proof of Stake for Blockchain Consensus, in: Proceedings of the 3rd International Conference on Blockchain Technology (ICBCT'21), 2021.

- [58] Ethereum.org, Rewards and Penalties in Proof-of-Stake, <https://ethereum.org/en/developers/docs/consensus-mechanisms/pos/rewards-and-penalties> (Erişim tarihi: 30 Mart 2025).
- [59] V. Buterin, Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform, 2014.
- [60] A.M. Antonopoulos, G. Wood, Mastering Ethereum: Building Smart Contracts and DApps, O'Reilly Media, 2018.
- [61] Anonim, Exploring Ethereum's Genesis Block, <https://medium.com/@mantri-nodes/exploring-ethereums-genesis-block-2d28fd282fb8> (Erişim tarihi: 31 Mart 2025).
- [62] Wikipedia, Ethereum, <https://ru.wikipedia.org/wiki/Ethereum> (Erişim tarihi: 31 Mart 2025).
- [63] D.G. Wood, Ethereum Yellowpaper, <https://ethereum.org/content/developers/tutorials/yellow-paper-evm/yellow-paper-berlin.pdf> (Erişim tarihi: 31 Mart 2025).
- [64] Namecoin.org, <https://www.namecoin.org/resources/whitepaper/> (Erişim tarihi: 21 Şubat 2025).
- [65] Litecoin, <https://litecoin.com/> (Erişim tarihi: 25 Şubat 2025).
- [66] Ripple, <https://xrpl.org/> (Erişim tarihi: 22 Şubat 2025).
- [67] Ripple, <https://ripple.com/> (Erişim tarihi: 22 Şubat 2025).
- [68] S.N. King, PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake, 2012.
- [69] Peercoin, <https://www.peercoin.net/> (Erişim tarihi: 22 Şubat 2025).
- [70] Dogecoin, <https://dogecoin.com/> (Erişim tarihi: 23 Şubat 2025).
- [71] N. v. Saberhagen, CryptoNote v 2.0, 2013.
- [72] Dash, <https://docs.dash.org/en/stable/docs/user/introduction/about.html> (Erişim tarihi: 6 Mart 2025).
- [73] Zcash, <https://z.cash/> (Erişim tarihi: 6 Mart 2025).
- [74] Steem Topluluğu, Steem: An incentivized, blockchain-based, public content platform, 2017.
- [75] Steem, <https://steem.com/> (Erişim tarihi: 7 Mart 2025).

- [76] Cardano, <https://docs.cardano.org/> (Eriřim tarihi: 3 Mart 2025).
- [77] BNB Chain, <https://www.bnbchain.org/en> (Eriřim tarihi: 7 Mart 2025).
- [78] S. Ellis, A. Juels, S. Nazarov, ChainLink: A Decentralized Oracle Network, 2017.
- [79] Chainlink, <https://chain.link/> (Eriřim tarihi: 7 Mart 2025).
- [80] D.G. Wood, Polkadot: Vision for a Heterogeneous Multi-Chain Framework, 2025.
- [81] Polkadot, <https://polkadot.com/> (Eriřim tarihi: 11 Mart 2025).
- [82] A. Yakovenko, Solana: A New Architecture for a High Performance Blockchain v0.8.13, 2017.
- [83] Solana, <https://solana.com/tr> (Eriřim tarihi: 12 Mart 2025).
- [84] Mina Protocol, <https://minaprotocol.com/lightweight-blockchain> (Eriřim tarihi: 25 Mayıs 2025).
- [85] IOTA, <https://www.iota.org/learn/intro> (Eriřim tarihi: 25 Mayıs 2025).
- [86] Avalanche, Avalanche Whitepapers, <https://www.avalabs.org/whitepapers> (Eriřim tarihi: 17 Nisan 2025).
- [87] Avalanche, <https://www.avalabs.org/> (Eriřim tarihi: 13 Mart 2025).
- [88] Guardtime, <https://guardtime.com/> (Eriřim tarihi: 21 Mart 2025).
- [89] IBM Food Trust, <https://www.ibm.com/docs/tr/food-trust?topic=overview> (Eriřim tarihi: 21 Mart 2025).
- [90] e-Estonia, <https://e-estonia.com/how-did-estonia-carry-out-the-worlds-first-mostly-online-national-elections/> (Eriřim tarihi: 22 Mart 2025).
- [91] Shenzhen Government Online, https://www.sz.gov.cn/en_szgov/govt/officials/qinwz/ (Eriřim tarihi: 22 Mart 2025).
- [92] U.S. Securities and Exchange Commission, <https://www.sec.gov/newsroom/speeches-statements/gensler-statement-spot-bitcoin-011023> (Eriřim tarihi: 22 Mart 2025).
- [93] A.M. Antonopoulos, Mastering Bitcoin, 2014.
- [94] Türkiye Cumhuriyet Merkez Bankası, Dijital Türk Lirası Birinci Faz Deęerlendirme Raporu, 2023.

- [95] D.W. Arner, D.A. Zetsche, R.P. Buckley, J. Kirkwood, The Financialization of Crypto: Lessons from FTX and the Crypto, UNSW Law & Justice Research Series, (2022–2023) 14.
- [96] Bitnodes, <https://www.bitnodes.io> (Erişim tarihi: 27 Kasım 2024).
- [97] U.S. Securities and Exchange Commission, <https://www.sec.gov/newsroom/speeches-statements/gensler-statement-spot-bitcoin-011023> (Erişim tarihi: 22 Mart 2025).

