T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**Doktora Tezi**

# AKILLI ORTAMLARDA BLOCKCHAIN TABANLI KİMLİK DOĞRULAMA SİSTEMİNİN GELİŞTİRİLMESİ

**Mohammed ALSADI**

**Enformatik Anabilim Dalı**

**Enformatik Programı**

**DANIŞMAN**
**Prof. Dr. Sevinç GÜLSEÇEN**

**II. DANIŞMAN**
**Dr. Öğr. Üyesi Büşra ÖZDENİZCİ KÖSE**

**Mart, 2020**

**İSTANBUL**

Bu çalışma, 2.03.2020 tarihinde aşağıdaki jüri tarafından Enformatik Anabilim Dalı, Enformatik Programında Doktora tezi olarak kabul edilmiştir.

**Tez Jürisi**

Prof. Dr. Sevinç GÜLSEÇEN(Danışman)
İstanbul Üniversitesi
Enformatik Bölümü

Doç. Dr. Seda TOLUN
İstanbul Üniversite
İşletme Fakültesi

Prof. Dr. Hacı Ali MANTAR
Gebze Teknik Üniversitesi
Mühendislik Fakültesi

Prof. Dr. Vedat COŞKUN
Beykent Üniversitesi
Mühendislik-Mimarlık Fakültesi

Doç. Dr. Çiğdem EROL
İstanbul Üniversitesi
Enformatik Bölümü

20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

# ÖNSÖZ

Doktora eğitimimde danışmanlığımı üstlenen Prof. Dr. Sevinç GÜLSEÇEN hocama bu tez çalışmasını gerçekleştirmem için verdiği destek ve sağladığı önemli katkılar için teşekkürlerimi sunarım.

Yüksek lisans ve doktora eğitimim boyunca ihtiyaç duyduğum her anımda yanımda olan, çalışmalarıma her zaman ve her şekilde destek veren, üzerimde büyük emeği olan Prof. Dr. Vedat COŞKUN hocama teşekkürlerimi sunarım.

Tez izleme komitesinde yer alan değerli Prof. Dr. Hacı Ali MANTAR hocama ve değerli Doç. Dr. Seda TOLUN hocama doktoram boyunca sunduğu katkılar için teşekkür ederim.

Birlikte çalıştığım değerli Öğretim Üyesi Dr. Büşra ÖZDENİZCİ KÖSE hocama çalışmalarımın her aşamasında verdiği katkılar için teşekkür ederim.

Eğitim hayatım boyunca, desteklerini esirgemeyen ve bu tez çalışmasının gerçekleşmesini sağlayan sevgili aileme, eşim Hadil ALFARRA ile oğlum Yusef ALSADI'ye sevgilerimi sunarım.

Mart, 2020. Mohammed ALSADI

# İÇİNDEKİLER

**Sayfa No**

# ŞEKİL LİSTESİ

**Sayfa No**

# TABLO LİSTESİ

**Sayfa No**

# SİMGE VE KISALTMA LİSTESİ

| Kısaltmalar | Açıklama |
|---|---|
| **BG** | : Bizans Generalleri (Byzantine Generals) |
| **DApp** | : Dağıtık Uygulama (Distributed Application) |
| **DDoS** | : Dağıtık Hizmet Engelleme Saldırısı (Distributed Denial of Service) |
| **ECDSA** | : Eliptik Eğri Dijital İmza Algoritması (Elliptic Curve Digital Signature Cryptography) |
| **IoT** | : Nesnelerin İnterneti (Internet of Things) |
| **M2M** | : Makineler Arası İletişim (Machine to Machine communication) |
| **OTP** | : Tek Kullanımlık Şifre (One Time Password) |
| **P2P** | : Uctan Uca (Peer-to-Peer) |
| **PIN** | : Kişisel Kimlik Numarası (Personal Identification Number) |
| **SPV** | : Basit Ödeme Doğrulama (Simplified Payment Verification) |

# ÖZET

## DOKTORA TEZİ

### AKILLI ORTAMLARDA BLOCKCHAIN TABANLI KİMLİK DOĞRULAMA SİSTEMİNİN GELİŞTİRİLMESİ

**Mohammed ALSADI**

**İstanbul Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Enformatik Anabilim Dalı**

**Danışman : Prof. Dr. Sevinç GÜLSEÇEN**

**II. Danışman : Dr. Öğr. Üyesi Büşra ÖZDENİZCİ KÖSE**

Günümüzün akıllı uygulamaları büyük oranda veri depolamak, işlemek, güvence altına almak ve dağıtmak için buluta bağımlıdır. Klasik İstemci-Sunucu mimarisinden bulut tabanlı mimariye geçiş, işletme maliyeti düşük ve ölçeklenebilir hizmetler sunmak yanında mobil cihazların (giyilebilir ve akıllı telefonlar gibi) verilere erişilmesine izin vermek gibi bir dizi avantaj sağlayan büyük bir değişim paradigması olarak kabul edilmektedir. Hızlı gelişime katkıda bulunmasına rağmen, merkezi mimarinin performans, güven ve güvenlik ile ilgili bazı dezavantajları vardır.

2008 yılında Satoshi Nakamoto tarafından sunulan ve Bitcoin teknolojisinin temelini oluşturan Blockchain; finans, tedarik zinciri, akıllı şebekeler, enerji ve diğerleri dahil olmak üzere çeşitli alanlarda yoğun ilgi görmektedir. Blockchain, kriptografik özetleme (hashing) yöntemi ile birbirine bağlanarak düzenlenmiş bir dizi kayıttan oluşan, bloklar halinde dağıtık bir şekilde verileri saklayan, kurcalamaya dayanıklı bir defterdir. Yeni işlemler veya

kayıtlar oluştukça, Blockchain ağının dayandığı defter büyümeye devam eder. Blockchain'in dağıtık doğası, tek bir paydaş tarafından defterin kontrol edilmediği, aksine tüm paydaşlar tarafından ortaklaşa bir konsensüs yaklaşımı ile kayıtların gerçekliğinin onaylandığı anlamına gelmektedir. Blockchain teknolojisinin sağladığı bu özellikler, merkezi bir paydaşa ihtiyaç duymadan, paydaşların kendi aralarında güvenilir bir ağ oluşturulmasını mümkün kılmaktadır.

Bu tez araştırmasının amacı, akıllı ortamlarda kullanılacak güvenilir bir Blockchain tabanlı kimlik doğrulama modeli tasarlamak ve geliştirmektir. Önerilen model, son kullanıcıların mevcut Blockchain ağına güvenli bir kimlik doğrulama yönetimiyle kayıt olmasına ve kayıtlı kullanıcıların çeşitli servislere ait bilgileri etkili bir erişim kontrol mekanizması ile sorgulayabilmesine imkan sağlamaktadır. Üst seviyede güvenlik sağlamak amacıyla, devlet kuruluşu bir paydaş olarak modele dahil edilerek ve son kullanıcıların kimliklerinin doğrulanması süreci tasarlanmıştır. Modelin diğer önemli bir parçası olan Akıllı Sözleşmeler ile sistem dinamikliği artırılmıştır. Bununla beraber doğası gereği şeffaf olması gereken Blockchain ağında, uygun bir erişim kontrol mekanizması tasarlanarak ağda bulunan katılımcıların işlem bilgilerini sorgulayabilmesi sağlanmıştır.

Mart 2020, 80 sayfa.

**Anahtar kelimeler:** Blokchain, akıllı sözleşmeler, akıllı ortamlar, IoT, kimlik doğrulama, erişim kontrolü.

# SUMMARY

## Ph.D. THESIS

### DEVELOPMENT OF BLOCKCHAIN BASED AUTHENTICATION SYSTEM IN SMART ENVIRONMENTS

**Mohammed ALSADI**

**İstanbul University**

**Institute of Graduate Studies in Sciences**

**Department of Informatics**

**Supervisor : Prof. Dr. Sevinç GÜLSEÇEN**

**Co-Supervisor : Assist. Prof. Dr. Büşra ÖZDENİZCİ KÖSE**

Today's smart applications depend heavily on cloud for storing, processing, securing and distributing data. The move from classical Client-Server architecture toward cloud-based is considered a big shift paradigm that leads to a number of advantages such as providing more scalable services with less operation cost and allowing usage of mobile devices (wearable and smartphones) to access data. Despite contributing to rapid development centralized architecture still has some drawbacks related to performance, trust, and security.

Blockchain is a tamper-resistant ledger, which consists of a set of records organized and stored in blocks interconnected to each other through cryptographic hashing approach. Blockchain-ledger- will continue to grow as new transactions are created. The distributed nature of Blockchain means that the ledger is not controlled by a single entity, but rather a consensus approach by the participant entities, confirming the authenticity of the records. These features

of Blockchain technology enable creation of trusted network among entities without the need for a central asset.

The aim of this thesis research is to design and develop a trustworthy Blockchain based authentication model that will be used in smart environments. The proposed model allows end users to register with the current Blockchain network with secure authentication approach and allow registered users to query information about various services with an effective access control mechanism. In order to provide high-level security, the government agency is included in the model as an entity and the end-user authentication process is designed. Another important part of the model is Smart Contract, which increases system dynamicity. However, in Blockchain network, which should be transparent by nature, an appropriate access control mechanism is designed to enable participants in the network to query transaction information.

March 2020, 80 pages.

**Keywords:** Blockchain, smart contract, smart environments, IoT, authentication, access control.

# 1. GİRİŞ

## 1.1. TEZ KONUSU VE ÖNEMİ

Internet ve mobil teknolojiler alanındaki hızlı gelişmeler ile İnternet'e bağlanabilme özelliği olan cihazlar (akıllı telefon, algılayıcı, tetikleyici ve benzeri cihazlar) hayatımızın her alanında yer almaya başlamıştır ve bu cihazların sayısı giderek artmaktadır. Akıllı ortam (Smart environment) algılayıcılar (sensors), tetikleyiciler (actuators) ve bilgi işlem bileşenleri ile kapsamlı bir şekilde donatılmış bir gerçek dünyaya ait bölge veya alandır [1]. Akıllı ortamlar, cihazların birbirleriyle bağlanması ve cihazları uzaktan veya otomatik kontrol edilmesi yoluyla çeşitli katma değerli uygulamalar sunmayı ve insanların hayatını kolaylaştırmayı amaçlamaktadır. Cihazlar birbiriyle iletişim kurmak için farklı teknolojiler kullanır; çevrenin veya insanların durumu hakkında anlamlı veriler elde ederek bilinçli modeller oluşturur ve verileri paylaşır; ve hatta dış kaynaklardan kablolu veya kablosuz İnternet iletişim altyapısı yoluyla bilgi alabilir veya bilgi gönderebilir. Bu kapsamda Akıllı ortamlara, akıllı endüstri alanında akıllı ofisler [2], tedarik zinciri takibi [3], endüstriyel otomasyon [4] ve benzeri uygulamalar; akıllı evler alanında yaşlı ve engelli kişilere yönelik uygulamalar [5]; sağlık alanında özerk hasta izleme ve yardım etme [6] ve benzeri uygulamalar; kentsel planlama alanında akıllı şehirler [7] ve benzeri uygulamalar örnek olarak verilebilir.

Nesnelerin İnterneti (IoT) paradigmasının yaygınlaşması ve fiziksel IoT cihazlarının artması Akıllı Ortamların gelişiminde önemli bir rol oynamaktadır [8-10]. Her ne kadar farklı ağ türleri, yenilikçi iletişim protokolleri, veri analitiği, makine öğrenmesi algoritmaları gibi Akıllı Ortamların gelişmesini sağlayan temel teknolojiler, çeşitli araştırma toplulukları tarafından yoğun ilgi görse de; söz konusu ortamların ve sistemlerin güvenilirliği, gizliliği ve şeffaflığı tartışılmakta olan önemli konulardır. Akıllı ortamlardaki cihazların üreteceği hassas verilerin işlenmesi, hassas verilerin saklanması, anlık olarak sunucudan alınması, ağ üzerinden transfer edilmesi ve benzeri işlemler düşünüldüğünde, bu ortamlarda kullanılabilecek merkezi (centralized) bir sistem mimarisinin önemli sıkıntılar yaratacağı öngörülmektedir.

Akıllı ortamlardaki cihazlar, esas olarak kablosuz bağlantılar yoluyla birbirleriyle iletişim kurduklarından güvenlik gereksinimleri sağlanmalıdır. Akıllı ortamlarda güvenlik gereksinimi,

diğer tüm bilgisayar sistemlerinde güvenlik gereksinimi ile aynı anlamı ifade etmektedir: bilgilerin çalınmaması, değiştirilmemesi veya buna erişimin engellenmemesi demektir [11]. Bu kapsamda akıllı ortamların ölçeklenebilirliği, kimlik doğrulama, entegrasyon, veri şifreleme, erişim kontrolü gibi temel güvenlik gereksinimleri bulunmaktadır [12].

Merkezi yapıya sahip olan sistemlerin önemli sıkıntılarından bir tanesi Tek Nokta Hatası (Single Point of Failure) problemidir [13]. Sistem kontrolünün tek bir sunucuya bırakılması durumunda, sunucunun çalışmasını engelleyecek bir sorunun oluşması sistemin çökmesine yol açmaktadır. Merkezi yapıda olan sistemler için sorun yaratan diğer önemli bir konu da verilerin gizliliğidir [14]. Hassas bilgilerin merkezi bir yapı tarafından yönetilmesi durumu, insanların mahremiyetini yüksek risk altına almaktadır. Her sistem için güvenlik bir önemli bir konudur; merkezi yapıya sahip olan sistemlerde verilerin tek bir yerde saklaması ve merkezi tek bir sunucu üzerinden tüm işlemlerin gerçekleştirilmesi konuları, bu sistemlerin DoS ve DDos gibi saldırı türleri için kolay bir hedef olmasını sağlamaktadır [15]. Bu kapsamda merkezi olmayan, diğer adıyla dağıtık bir yapıya sahip sistemlere duyulan ihtiyaç giderek artmaktadır.

Son yıllarda giderek önem kazanan Blockchain (Blok Zinciri) teknolojisi, bilgi ya da olayları (gerçekleri) gerçekleştiği zaman bilgisi (timestamp) ile birlikte sonradan değiştirilemez şekilde kayıt altına almayı mümkün kılan önemli bir teknolojidir. Dağıtık bir veri ağı üzerinde faaliyet gösterme olanağı sunan Blockchain teknolojisi, verileri güvenli ve kalıcı olarak saklama, hızlı transfer etme ve erişilebilir olma ve benzeri avantajlara sahiptir. Blockchain, birbirine güvenmeyen kullanıcıların üçüncü bir tarafa ihtiyaç duymaksızın değiştirilmez ve reddedilemez bir veri üzerinde anlaşabilmelerini sağlamaktadır.

Blockchain teknolojisine bağlı olan ağların üç farklı tipi bulunmaktadır: *Özel Blockchain ağı (Private Blockchain network), Açık Blockchain ağı (Public Blockchain network), ve Konsorsiyum Blockchain ağı (Consortium Blockchain network).* Özel Blockchain ağlarında, tüm işlemler ağ sahibi tarafından kontrol edilmektedir. Verilere kimin erişebileceği, kimin işlem yapabileceği ve kimin düzenleme hakkına sahip olabileceği konularına ağ sahibi karar vermektedir. Açık Blockchain ağlarında, ağa katılım olanağı herkese tanımlamaktadır. Bu ağ türünde yapılan her işlem, ağdaki tüm kullanıcılar tarafından gözlemlenmektedir. Bitcoin, bu ağ türünün en bilinen örneğidir. Bazı kaynaklarda federe Blockchain olarak da adlandırılan

Konsorsiyum Blockchain ağı ise, önceden seçilmiş bazı düğümler tarafından kontrol edilen halka yarı açık, kısmi izin verilen bir sistemdir.

Blockchain teknolojisinin en büyük özelliği merkezi olmayan, dağıtık bir doğrulama sistemine sahip olmasıdır. Yapılan her işlem, Blockchain ağına eklenmeden belirli kullanıcılar, diğer adıyla doğrulayıcılar tarafından onaylanması ve doğrulanması gerekmektedir. Blockchain teknolojisi, dijital dönüşümün yaşandığı en etkili alanlardan biri olarak gösterilmektedir. Sağladığı özellikler sayesinde, Blockchain teknolojisi sadece finans sektörü ile sınırlı kalmayıp enerji, akıllı ortamlar, tarım, tedarik zinciri, sağlık ve benzeri diğer alanlarda da kullanılmaya başlanmıştır.

## 1.2. TEZİN AMACI

Tez çalışmasının amacı, akıllı ortamlarda kullanılabilecek olan Konsorsiyum Blockchain tabanlı bir kimlik doğrulama modeli tasarlanması ve geliştirilmesidir. Önerilen model sayesinde birbirine güvenmeyen aktörler arasında, dağıtık bir yapı üzerinden ağ oluşturmaları ve reddedilemez kayıtların oluşturulması sağlanacaktır. Geliştirilecek modelin, dinamik bir yapıda olabilmesi için Akıllı Sözleşme (Smart Contract) altyapısı geliştirilerek aktörlere erişim izinleri tanımlanacak, ve aktörlerin birbirlerine güvenli bir şekilde erişmesi ve ulaşması sağlanacaktır.

Tez çalışması beş ayrı bölümden oluşmaktadır: "Giriş" bölümünden sonra ikinci bölüm "Genel Kısımlar", üçüncü bölüm "Malzeme ve Yöntem", dördüncü bölüm "Bulgular" ve son olarak "Tartışma ve Sonuç" bölümü gelmektedir.

"Genel Kısımlar" bölümünde, konuyla ilgili yapılan akademik araştırmalar ve literatür taraması sonuçları yer almaktadır. Akıllı ortamların ve güvenlik gereksinimleri ile ilgili bilgiler, Blockchain teknolojisinin mimarisi, sistemleri ve platformları ile ilgili bilgiler, akıllı sözleşmelerin detaylar paylaşılmıştır.

"Malzeme ve Yöntem" bölümünde, tez çalışmasında izlenen yöntem, geliştirilecek olan Blockchain tabanlı kimlik doğrulama modellin aktörleri, modelin mimarisinin sağlaması gereken güvenlik gereksinimleri, sistem gerçekleşme aşamaları paylaşılmıştır.

"Bulgular" bölümünde, tasarlanan kimlik doğrulama modelin ve erişim mekanizmasının geliştirme çalışmaları paylaşılmıştır.

"Tartışma ve Sonuç" bölümünde, tezde yapılan çalışmalar özetlenmiştir; geliştirilen Blockchain tabanlı kimlik doğrulama modelin katkıları sunulmuştur.

# 2. GENEL KISIMLAR

## 2.1. AKILLI ORTAMLAR VE GÜVENLİK GEREKSİNİMLERİ

Günümüzde içinde bulunduğumuz Endüstri 4.0 dönemi, yeni kavramların doğmasını ve hayatımıza dokunan katma değerli servislerin gelişerek önem kazanmasını sağlamıştır. Makineleşme, otomasyon sistemleri, makineler arası iletişim (M2M) kavramlarının ötesinde, yeni iletişim ve ağ teknolojileriyle birbirleriyle iletişim kurabilen ve Internet'e bağlanabilen nesnelerin iletişimi ve bu iletişimin oluşturduğu verilerin yönetimi, güvenliği ve gizliliği konuları içeren yeni büyük bir ekosistemi, diğer adıyla Nesnelerin İnterneti (Internet of Things, IoT) paradigmasını doğurmuştur [16].

IoT paradigması, gerçek dünya nesnelerine algılama, uyarma, tetikleme ve benzeri yeteneklerin kazandırılarak, insan müdahalesi olmadan "Akıllı" nesnelerin aralarında iletişim kurmasını sağlayan bir dizi teknolojiyi kapsar [17]. IoT ağı ile birbiri ile iletişim kurabilen nesnelerden elde edilen veriler üzerinde veri analitiği ve makine öğrenmesi çalışmalarının uygulanmasıyla akıllı sağlık, akıllı şebeke, akıllı şehir, akıllı ev, akıllı tarım, akıllı- trafik, akıllı araç ve benzeri birçok "Akıllı Ortam" içerisinde katma değerli, yenilikçi servislerin sunulması sağlanabilmektedir [18].

IoT ekosisteminin büyümesiyle, akıllı ortam kavramı da giderek geliştirmekte ve çeşitlilik kazanmaktadır. "Akıllı Ortam" kavramı, yapılan bir çalışmada [19] şu şekilde tanımlanmaktadır: "Hayatımızın gündelik nesnelerine sorunsuz bir şekilde yerleştirilen ve sürekli bir ağ üzerinden bağlanan algılayıcılar, tetikleyiciler, ekranlar ve hesaplama unsurlarıyla zengin ve görünmez bir şekilde iç içe donatılmış fiziksel bir dünyadır." Diane ve diğerleri [20] göre akıllı ortam, insanların yaşamını daha rahat hale getirmek için sürekli olarak çalışabilen ve her türlü akıllı cihazı barındıran küçük bir dünyadır. Bu tanımda "akıllı" kavramı, özerk bir şekilde bilgi edinme ve uygulama yeteneği anlamına gelmektedir; "ortam" kavramı ise, çevremizi ifade etmektedir.

Yapılan bir başka çalışma [21] ise teknolojisi açıdan akıllı ortam, algılayıcılar (sensors), veri işleme ve tetikleyiciler (actuators) olmak üzere üç ana bileşenden oluşan bir ortam olarak tanımlanmaktadır ve bu bileşenler şu şekilde açıklanmaktadır:

- Algılayıcı: Fiziksel cihazlara gömülü olan ve duyma, algılama, görme ve benzeri yöntemlerle çevreden bilgi toplamak amacıyla kullanılan cihazlardır.
- Veri İşleme: Akıllı bir ortamın merkezi bileşenidir. Algılayıcıların topladığı verilerin, zaman içindeki değişikliklerini tespit etmek veya belirli bir zamanda belirli bir noktada belirli bir değişiklik olup olmadığını tespit etmek amacıyla işlenmesinden sorumlu olan bileşenidir.
- Tetikleyici: Veri işleme aşamasının çıktısına dayanarak çevreyi kontrol eden ve belli cihazlara, ortamlara veya kişilere geri bildirim sağlayan bir çevre kontrolüdür.

Diğer taraftan Internet ve ağ teknolojileri alanında yaşanan gelişmeler, IoT ekosistemlerinde ve akıllı ortamlarda kullanıcı etkinliklerinin izlenmesi, hassas verilerin toplanması, verilerin gizliliği ve hatta güvenli kimlik doğrulama ve erişim kontrolü ihtiyacı konularını da gündeme getirmektedir. Örneğin, 21 Ekim 2016 tarihinde bir Alan Adı Sistemi (Domain Name System, DNS) sağlayıcısı olan Dyn firması, Internet'i oluşturan temel altyapıların ciddi derecede çökmesine yol açan bir dağıtık hizmet engelleme (DDoS) saldırısı yaşamıştır; bu saldırıdaki önemli nokta ise, saldırıyı başlatan bilgisayarların yazıcılar, web kameraları, yerleşim ağ geçitleri ve bebek monitörleri ve benzeri küçük IoT cihazlar olmasıdır [22].

Kimlik doğrulaması, IoT ekosistemleri için en temel gereksinimlerinden biri olarak kabul edilmektedir [23]. IoT ekosistemlerinde kimlik doğrulama ve erişim kontrol mekanizmaları çeşitli çalışmalarda araştırılmıştır [24-26]. Kimlik doğrulama ve erişim kontrolü teknolojileri, bilgisayar ağlarındaki güvenlik ve gizlilik sorunlarını ele alan merkezi unsurlar olarak bilinmektedir. Yapılan bir çalışma [24] eşler arası kimlik doğrulama yönteminin, IoT'nin verimli bir şekilde güvenliğini sağlamak için kullanılabilecek bir çözüm olduğunu, ancak eşler arasındaki karşılıklı güven problemini çözmeden, bu çözümün başarılı olamayacağı vurgulanmaktadır.

## 2.2. BLOCKCHAIN TEKNOLOJİSİ

Blockchain, Internet'ten sonra dünyadaki birçok sektörü değiştirebilme potansiyeline sahip olan en önemli teknolojilerinden biri olarak görülmektedir. 2008 yılında Satoshi Nakamoto tarafından "Bitcoin: A Peer-to-Peer Electronic Cash System" adlı makalesinde, Blockchain

teknolojisi yayınlanarak tüm dünyaya duyurulmuştur [27]. Bu çalışmaya göre, Blockchain teknolojisi, kripto para olarak kullanılan ilk para birimi, Bitcoin'in altyapısını oluşturmaktadır.

Blockchain, eşler arası ağ üzerinde yer alan katılımcıların mutabakat sağlanmasına dayalı, değer taşıyabilecek varlıkların transfer edilmesini sağlayan dağıtık bir veri tabanı sistemidir [28]. Bu teknoloji sayesinde, eşler arası yapılan kayıtlar herhangi bir merkezi otoriteye gerek kalmadan, tamamen dağıtık bir şekilde yapılmaktadır [29].

Blockchain teknolojisi değiştirilemez, dağıtılmış, bir şekilde çalışan ve matematiksel olarak ispat edilmiş, bir şekilde güvenli veri içeren bir veri tabanıdır [30]. Blockchain, birbiri ile bağlanmış bloklardan oluşan bir listedir ve yeni bloklar eklendikçe bu liste büyümeye devam etmektedir. Bu teknoloji sayesinde bir hareketin veya bir kaydın (transaction) merkezi bir otorite olmaksızın kaydı tutulabilmekte, bileşenler arasında bu sayede bağımsız olarak bir güvenli haberleşme olabilmektedir.

Yapılan çalışmalara [31-33] göre, Blockchain teknolojisinin en temel özellikleri şu şekildedir:

(a) **Değiştirilemez:** Blockchain ağına eklenmiş herhangi bir kayıt artık değiştirilemez özelliğe sahiptir. Yeni bir kayıt eklenmek istendiğinde önce içeriğin matematiksel olarak özet bilgisi (hash) alınıyor ve her kayıt bir önceki kaydın özet bilgisini barındırarak ekleniyor. Böylece, herhangi bir kayıtın içeriği değiştirilmek istendiği zaman veya araya başka bir kayıt eklenmeye çalışıldığında, Blockchain sistemi bu işlemi yapısı gereği onaylamamaktadır.

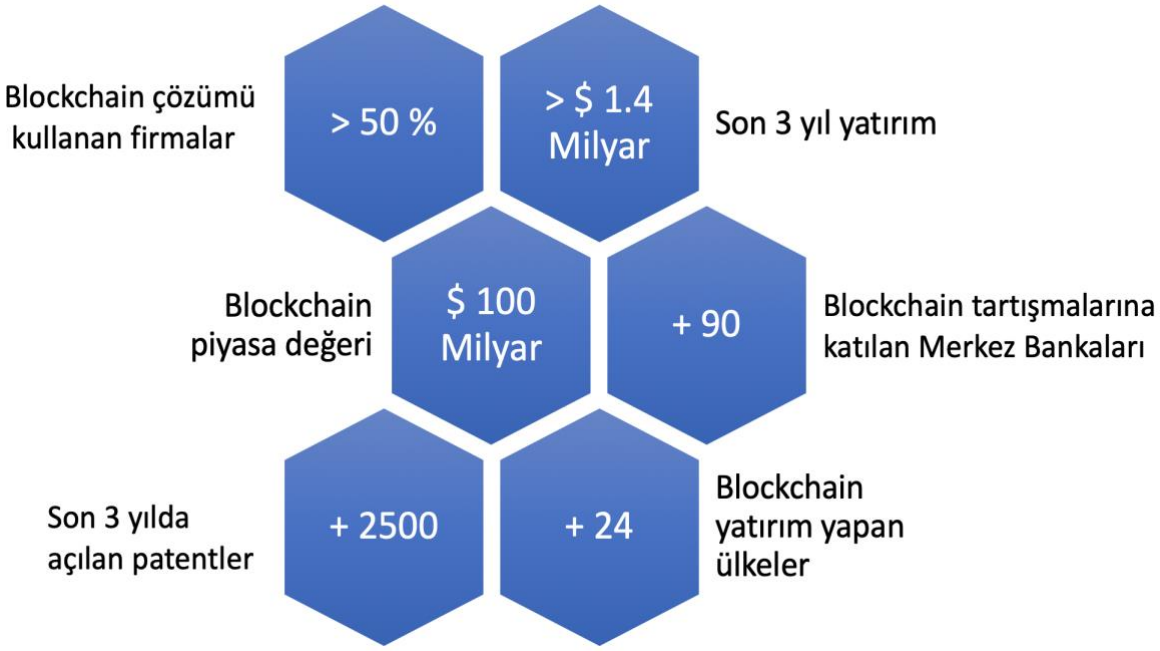
- **Dağıtık**: Blockchain teknolojisini tanımlayan en önemli özelliklerinden biridir. Dağıtık özelliği ile merkezi bir yapıyı değil, aksine herkesin sahip olabileceği bir yapıyı ifade edilmektedir. Yapısı gereği, bir kaydın veya işlemin doğruluğu, Blockchain ağı üzerinde bulunan belirli sayıda noktadan, diğer adıyla düğümden onay alınarak kabul edilmektedir.
- **Kriptografik Olarak Güvenli**: Kullanılan özetleme fonksiyonları (hash functions) sayesinde, Blockchain ağının içeriği güvenli bir şekilde korunmaktadır.

Güvenlik gereksinimleri olarak kayıtların bütünlüğü, kullanılabilirliği, gizliliği, hata toleransı ve hesaplama zamanı açılarından, merkezi veri tabanı sistemleri, dağıtık veri tabanı sistemleri ve Blockchain teknolojisi arasındaki farklar Tablo 2.1'de paylaşılmaktadır [34].

**Tablo 2.1:** Blockchain ile veri tabanı arasındaki farkı.

| **Güvenlik Özellikleri** | **Blockchain** | **Merkezi Veri Tabanı** | **Dağıtık Veri Tabanı** |
|---|---|---|---|
| Kayıtların Bütünlüğü | Yüksek | Orta | Orta |
| Kullanılabilirliği | Yüksek | Düşük | Orta |
| Gizlilik | Düşük | Yüksek | Orta |
| Hata Toleransı | Yüksek | Düşük | Yüksek |
| Hesaplama Zamanı | Düşük | Yüksek | Orta |

Günümüzde, Blockchain teknolojisinin sahip olduğu ve sağladığı bu özellikler sayesinde, finans sektörünün yanı sıra eğitim, enerji, sağlık ve hatta birçok akıllı ortam geliştirme çalışmalarında ilgi görmeye başlamıştır [35-37].




**Şekil 2-1**: Sayılar ile Blockchain [38-40].

### 2.2.1. Blockchain Mimarisi

Karmaşık bir teknoloji olarak görülen Blockchain sistemleri, belirli bir takım bileşenlerden oluşmaktadır: düğümler (nodes), kayıtlar (transactions), bloklar (blocks), kriptografi ve uzlaşma protokolleri (consensus protocols).

**(A) Düğümler**

Blockchain ağına bağlanan ve birbirleriyle iletişim kurması hedeflenen her bir bilgisayar veya cihaz, bir düğüm (node) olarak kabul edilir. Blockchain ağında sağlanan Uçtan Uca (Peer-to-Peer, P2P) bağlantıda düğümler eşit kabul edilse de, destekledikleri işlevselliğe bağlı olarak Blockchain ağı üzerinde bu düğümler farklı roller üstlenebilirler. Blockchain ağında, düğümlerin alabilecekleri roller şu şekildedir: Yönlendirme, Blockchain Veri Tabanı, Madencilik (Doğrulama) ve Cüzdan (Wallet) hizmetleridir [41]. Tüm düğümler ağa katılmak için yönlendirme işlevini içermektedir:

- Yönlendirme fonksiyonu sayesinde bir düğüm, diğer düğümleri keşfedebilir; diğer düğümlere bağlanabilir; kayıtları ve blokları yayabilir.
- Blockchain veri tabanı fonksiyonu sağlayan düğümler, Blockchain'in eksiksiz ve güncel kopyasını tutar. Blockchain veri tabanı fonksiyonu genelde tam düğüm tarafından desteklenmektedir.
- Doğrulama rolüne sahip düğümler, uzlaşma algoritmasını çözmek için çalışır ve yeni bloklar oluşturmayı sağlar.
- Cüzdan hizmeti ise, kullanıcının sahip olduğu bitcoin'i görüntülemek, saklamak ve işlemek için kullanılır.

Yapılacak işlevlere göre, çeşitli Blockchain düğümleri bulunmaktadır. Tüm düğüm türleri yönlendirme rolünü sağlamaktadır. Başlıca düğüm türleri tam düğüm (full node), hafif düğüm (light node), Basit Ödeme Doğrulama (Simplified Payment Verification, SPV) düğümü, madenci düğümü (miner node) olarak ifade edilebilir [41]:

- Tam düğümler, oluşturulan tüm blokları kapsayıp, Blockchain ağı geçmişinin bir kopyasını içermektedir.

- Hafif Düğüm ve SPV düğümleri ise, yalnızca blok başlıklarını indiren ve dolayısı ile kullanıcılar için sabit disk alanı kazandıran cüzdanlardır. Satoshi Nakamoto'nun çalışmasında [27] açıklanan bir teknik olan SPV düğümü, hafif düğümlerin -tüm defteri indirmeden- bir işlemin Bitcoin'in Blockchain ağına dâhil edilmesini ve doğrulamasını sağlamaktadır.
- Madenci düğümleri (miner node), tam veya hafif düğüm olabilir. Zorlu bir hesaplama işlemini çözmek için özel donanımlar çalıştırarak, yeni bloklar oluşturmak için yarışan özel düğümlerdir.

**(B) Kayıtlar**

Kayıt (transaction), Blockchain ağına yayınlanan ve bloklar halinde toplanan varlık transfer işlemidir. Basit bir ifadeyle kayıt, bir varlığın sahibinin başka bir kişiye, ilgili varlığın devredilmesini onayladığı göstergesidir. Kayıtlar, düğümler tarafından doğrulanıp blok içinde açık bir biçimde tutulur. Dolayısıyla, kayıtlar Blockchain ağına katılanlar tarafından kolayca takip edilebilir.

Kayıtlar, Blockchain sisteminin en önemli parçasıdır. Diğer bileşenler, kayıtların oluşturulmasını, ağda yayılmasını, onaylanmasını ve sonunda Blockchain defterine yani dağıtık veri tabanına eklenmesini sağlamak için tasarlanmıştır [11]. Kaydın Yaşam Döngüsü (Transaction Life Cycle), kaydın oluşturulmasıyla başlar ve ardından işlem yapan düğüm tarafından imzalanmasıyla devam eder. İmzalanan kayıt, Blockchain ağında bulunan her düğüme erişerek yayınlanır. Her düğüm yeni oluşturulan kaydı doğrulayarak ilerletir. Son adımda, kayıt bir madenci düğümü (miner) tarafından hazırlanan aday bloğuna dahil edilir.

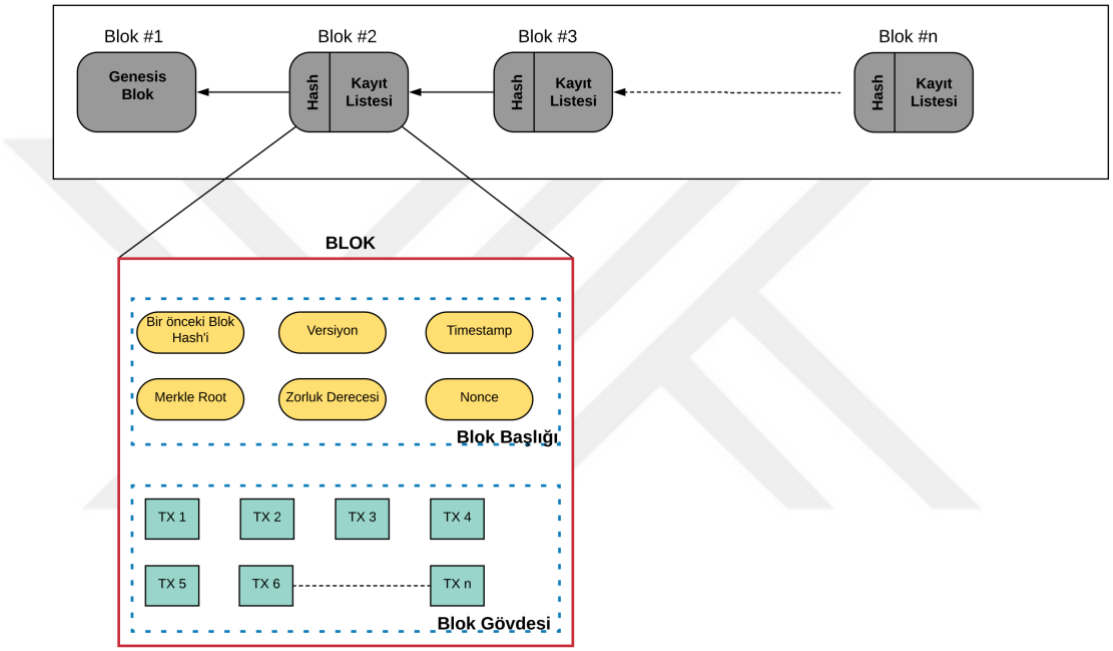
**(C) Bloklar**

Blockchain teknolojisinin adı, birbiri ile bağlanmış yani zincirlenmiş bloklardan gelmektedir. Düğümler tarafından oluşturulan kayıtlar, Blockchain veritabanına bir daha değiştirilmemek üzere blok haline getirilmektedir [42].

Blockchain sisteminde oluşturulan blokların genel yapısı Şekil 2.2'de verilmektedir. Bloklar, birbiri ile çift-SHA-256 (double-SHA-256) özetleme fonksiyonunu kullanarak bağlanır [43]. Her blokta iki kısım bulunmaktadır: blok başlığı (block header) ve blok gövdesi (block body). Blok başlığı, bloğun tanımlanması için kullanılmaktadır. Blok başlığında bulunması gereken

bilgiler Tablo 2.2'de verilmektedir. Blok gövdesi ise, bir bloğun içerdiği kayıtlardan (transactions) oluşmaktadır.

Blockchain ağının, ilk bloğu Akıllı Blok (Genesis Block) olarak adlandırılır. Bu blok, Blockchain ağı oluşturmak için gerekli olan, Blockchain ağının başlangıç davranışını tanımlayacaktır [44]. Genesis Block dışındaki her blok, bir önceki bloğun özet bilgisini içermektedir.



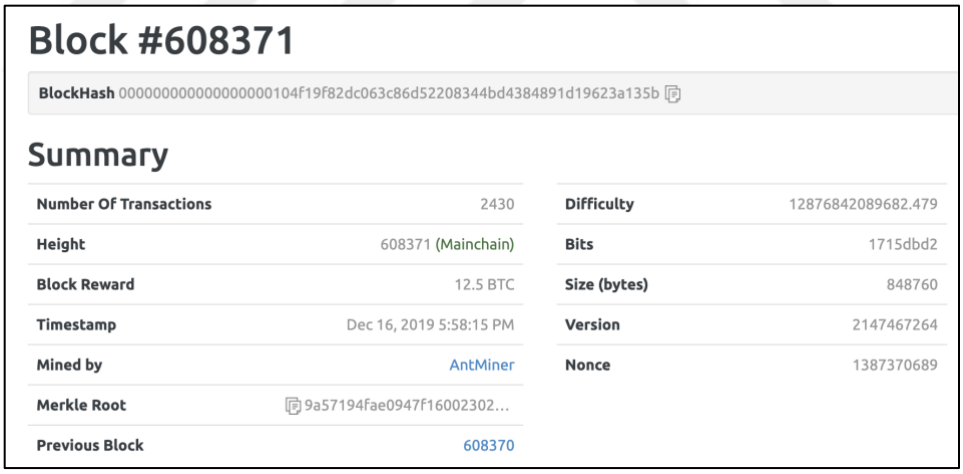

**Şekil 2-2**: Blockchain veri yapısı.

**Tablo 2.2:** Blok başlığının yapısı.

| **Alan** | **Uzunluğu** | **Açıklama** |
|---|---|---|
| Versiyon | 4 bayt | Blok doğrulama kurallarından hangisinin izleneceğini gösterir. |
| Bir Önceki Blok Hash'ı | 32 bayt | Bir önceki bloğun başlığının, çift-SHA-256 özetleme fonksiyonu çıktısıdır. |

**Tablo 2.2 (devamı):** Blok başlığın yapısı.

| | | |
|---|---|---|
| Merkle Kökü (Merkle Root) | 32 bayt | Blok kayıtlarının Merkle-Ağacı kökünün özet değeridir. |
| Zaman Damgası (Timestamp) | 4 bayt | Bloğun yaklaşık oluşturulma zamanıdır. |
| Zorluk Derecesi | 4 bayt | Blok için Proof-of-Work (PoW) algoritma zorluk hedefini gösterir. |
| Nonce | 4 bayt | Proof-of-Work (PoW) algoritması için bir sayaçtır. |




**Şekil 2-3:** Örnek bir Bitcoin bloğu.

Bir bloğun içerebileceği maksimum kayıt sayısı, blok boyutuna ve her kaydın boyutuna bağlıdır [45]. Blockchain, kayıtların onaylanmasını doğrulamak için asimetrik bir şifreleme mekanizması kullanmaktadır [46-48]. Şekil 2.4'de yeni bir kayıt doğrulama ve bloğa ekleme akışı gösterilmektedir. Bu süreç, bir örnekle şu şekilde açıklanabilir:

(1) Kullanıcı A, Kullanıcı B'ye bir para transferi gerçekleştirmek için kayıt oluşturur. Oluşturulan kayıt, diğer düğümler üzerinden Blockchain ağına yayılır.

(2) Yeni oluşturulan kayıt, farklı düğümler tarafından oluşturulan diğer kayıtlar ile birlikte havuzda bekletilir.

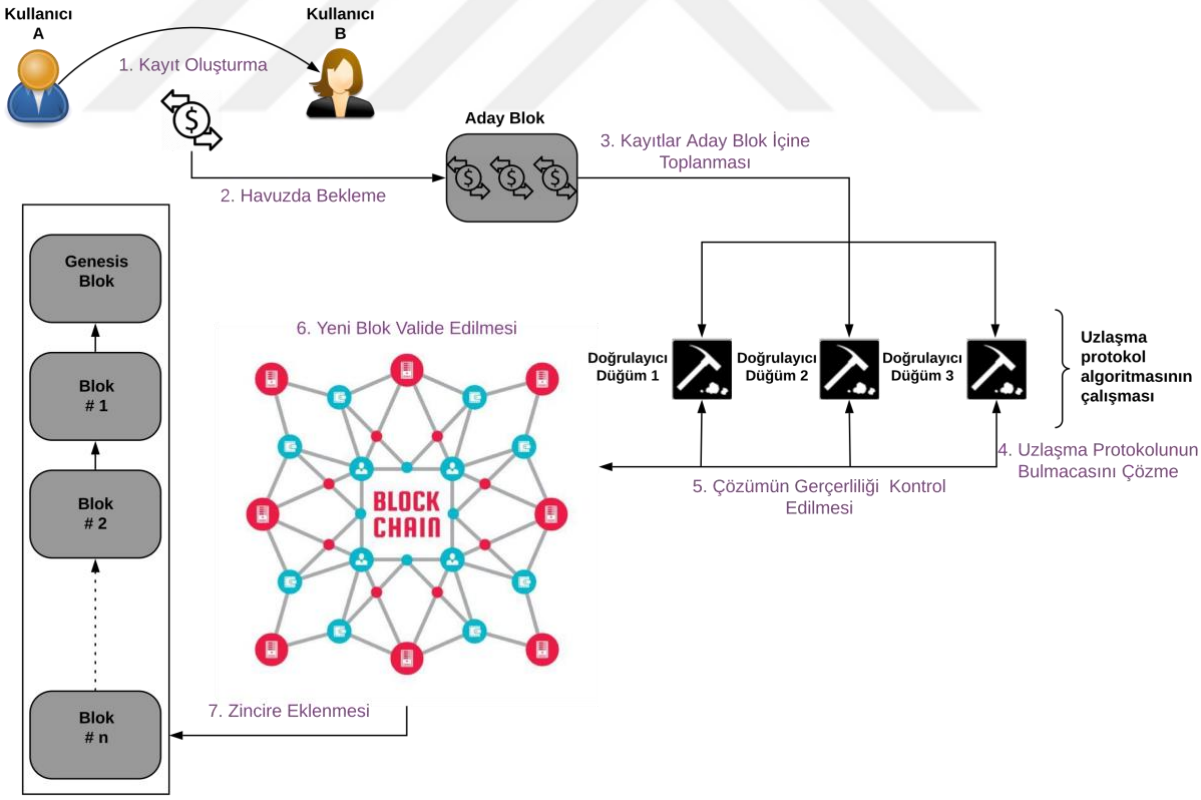
(3) Kayıtların Blockchain ağına dahil olması için, doğrulayıcı düğümler tarafından bir *aday blok* içinde ilgili kayıtlar toplanır.

(4) Doğrulayıcı düğümlerin *aday bloğu* zincire ekleyebilmesi için uzlaşma protokolünün bulmacasını çözmeleri gerekir. Bulmacayı çözmek geçerli bir nonce değeri bir düğüm tarafından bulunması demektir. Bulmacayı ilk çözen düğüm, *aday bloğu* zincire ekleyip bir ödül (kriptopara birimi) kazanır.

(5) Bulunan çözümün geçerliliği, diğer doğrulayıcı düğümler tarafından kontrol edilir. Çözümün geçerliliği kontrol edilmesi için bulunan nonce değeri ile bloğun başlığındaki diğer bileşenlerin özet değerini hesaplanarak hedef değerinden daha küçük olup olmadığına bakılır.

(6) Blockchain ağı düğümler tarafından *Aday blok* doğrulanır.

(7) *Aday blok* zincire, yani Blockchain ağına eklenir.




**Şekil 2-4:** Kayıt doğrulama ve yeni blok ekleme akışı.

### (D) Kriptografi

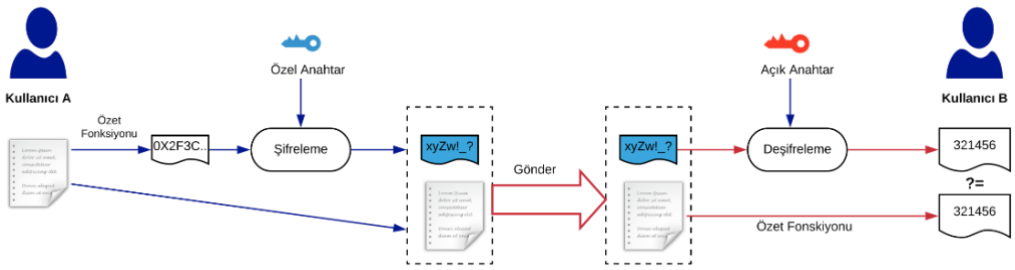
Blockchain teknolojsinin temeli, kriptografi üzerine inşa edilmiştir. Merkezi bir otorite olmadan bir sistemin güvenliğini sağlamak konusunda, kriptografi büyük bir öneme sahiptir. Blockchain sisteminde kullanılan kriptografi hem kayıtları oluşturanların kimliğini güvence altına alır, hem de geçmiş kayıtların tahrif edilmemesini sağlamaktadır.

Blockchain sisteminde kullanılan kriptografi araçları iki kategoride sınıflandırılmıştır: temel (primary) araçlar ve isteğe bağlı (optional) araçlar [49]. Temel araçlar, kriptografik özet fonksiyonları (cryptographic hash functions) ve dijital imzaları (digital signature) içerir. İkinci kategori ise, Blockchain temelli işlemlerin güvenliğini ve gizliliğini artırmak için isteğe bağlı olarak kullanılabilecek araçları içerir. Ring Signature, Zero-Knowledge Proof ve benzeri araçlar örnek verilebilir.

Blockchain teknolojisinin özünde, özetleme fonksiyonu (hash function) yer almaktadır. Özetleme fonksiyonu, farklı uzunluklardaki verilerin belirli bir uzunluğa dönüştürülmesini sağlayan ve aynı girdi ile her zaman aynı çıktıyı veren tek yönlü fonksiyondur [50]. İdeal bir kriptografik özetleme fonksiyonunda, herhangi bir veri girişi için kolayca bir özet değer üretebilir; ancak üretilen özet değer kullanarak orijinal veriyi elde etmek imkânsızdır [51]. Ek olarak, orijinal veride gerçekleşecek herhangi bir değişiklik, özet bilgisi (hash) değerinin farklılaşmasına neden olacaktır [52]. Son olarak, iki farklı veri girişinin aynı özet değer ile sonuçlanmaması gerekmektedir. Dolayısıyla, özetleme fonksiyonları veri bütünlüğünü güvenlik gereksinimini sağladığı için, Blockchain sistemlerinde büyük öneme sahiptir. Bitcoin, kriptografik para birimi uygulamalarında, SHA256 ve Keccak-256 özet fonksiyonları kullanılmaktadır [41].

Dijital imza, içeriğini ve gönderenin kimliğini doğrulamak için elektronik olarak iletilen bir belgeye eklenmiş olan dijital bir koddur [53]. Blockchain sisteminde, her kullanıcı bir çift özel ve açık anahtara sahiptir. Şekil 2.5'te, Blockchain ağında kullanılan bir dijital imza örneğini göstermektedir. Görüldüğü üzere dijital imzası, imzalama aşaması ve doğrulama aşaması olmak üzere iki aşamadan oluşur. Özel anahtar (Private Key), kayıtları imzalamak için kullanılır. Dijital imzalı bir kayıt, Blockchain ağının tamamına yayıldıktan sonra, ağdaki herkes tarafından görülebilen açık anahtar (Public Key) bilgisiyle erişilir. Blockchain sisteminde, en yaygın

olarak Eliptik Eğri Dijital İmza Algoritması (Elliptic Curve Digital Signature Cryptography, ECDSA) kullanılmaktadır [54].




**Şekil 2-5:** Dijital imza işlemi.

### (E) Uzlaşma Protokolleri

Blockchain sistemlerinde, birbirine güvenmeyen düğümler arasında nasıl uzlaşmaya varılacağı konusu, [55] yapılan çalışmasında da belirtilen Bizans Generalleri (Byzantine Generals, BG) probleminin bir dönüşümüdür. BG problemine göre, Bizans ordusunun bir bölümünü yöneten bir general, şehri kuşatmıştır; ordunun bazı diğer generalleri saldırmayı, bazıları da geri çekilmeyi tercih etmiştir. Generallerin sadece bir kısmı, şehre saldırmayı tercih ederse, saldırının başarısız olma ihtimali bulunmaktaydı. Bu nedenle, saldırı veya geri çekilme için bir anlaşmaya varmaları gerekmiştir.

Dağıtık (decentralized) ortamda, bir fikir birliğine nasıl ulaşılacağı önemli bir zorluktur. Blockchain sistemleri, dağıtık olarak bulunan düğümlerdeki defterlerin (ledger) -aynı olmasını sağlayan merkezi (centralized) bir düğüm olmadığından dolayı- tutarlı olmasını sağlamak için uzlaşma protokollerine, yani uzlaşma algoritmalarına (consensus algorithms) ihtiyaç duymaktadır [56].

Uzlaşma algoritması, Blockchain ağının tüm düğümlerine dağıtılmış defterlerin mevcut durumu hakkında ortak bir anlaşmaya ve fikir birliğine varmasını sağlayan bir algoritmadır. Bu şekilde, uzlaşma algoritmaları Blockchain ağında güvenilirliği sağlar ve dağıtık bir ortamda birbirini tanımayan veya birbirine güvenmeyen eşler (düğümler) arasında güven ortamı oluşturmaktadır [57].

Günümüzde, Blockchain sistemlerinde dört ana uzlaşma protokolü kullanılmaktadır [58]: Proof of Work (PoW) [27], Proof of Stake (PoS) [59], Practical Byzantine Fault Tolerance (PBFT)

[60] ve Delegated Proof of Stake (DPoS) [61]. Bunun dışında, Proof of Bandwidth (PoB) [62], Proof of Elapsed Time (PoET) [63], Proof of Authority (PoA) [64] gibi başka uzlaşma protokolleri de bazı özel Blockchain sistemlerinde kullanılmaktadır.

**Tablo 2.3:** Uzlaşma protokolleri arasında karşılaştırma [58, 65-68].

| **Özellik** | **PoW** | **PoS** | **DPoS** | **PBFT** |
|---|---|---|---|---|
| Ölçeklenebilirlik | Güçlü | Güçlü | Güçlü | Zayıf |
| Verimlilik (Kayıt/Saniye) | <100 | <1000 | <1000 | <2000 |
| Enerji tasarrufu | Hayır | Kısmi | Kısmi | Evet |
| Düğüm Kimliği Yönetimi (Node Identity Management) | Açık | Açık | Açık | İzinli |
| Örnek | Bitcoin | Peercoin | Bitshares | Hyperledger Fabric |

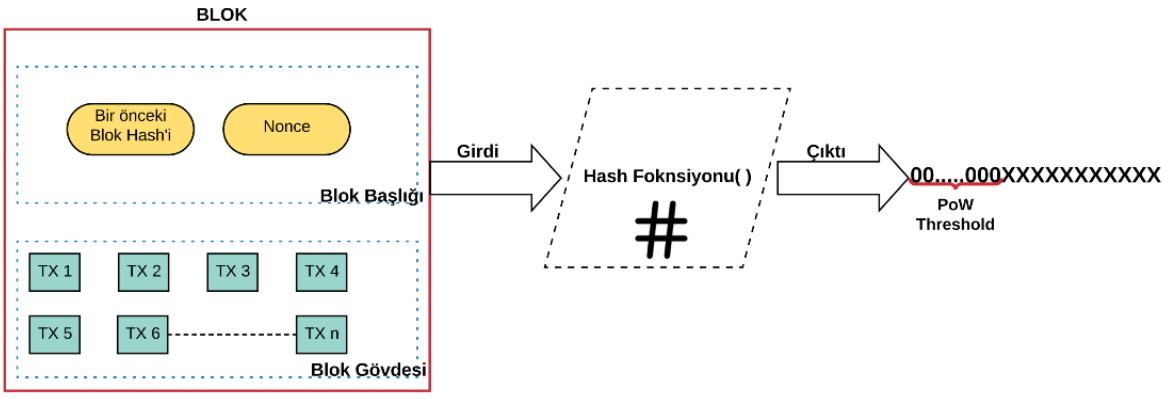
Bitcoin'de kullanılan Proof-of-Work algoritmasında, yeni bloğun oluşturulması düğümlerin hesaplama gücüne dayanmaktadır. Yeni bloğu ekleyip, BTC (Bitcoin para pirimi) ödülüne sahip olmak isteyen düğümlerin zor bir matematiksel bulmacayı çözmesi gerekmektedir. Bulmacayı çözen ilk düğüm, bu hakka sahip olur. Bitcoin'de bu bulmaca, HashCash bulmacası olarak tanımlanmaktadır [69].

Proof-of-Work algoritmasını kullanılan Blockchain ağının bir düğümü, yeni oluşturulacak olan aday bloğun başlığının özet (hash) değerini hesaplar [41]. Blok başlığı bir *nonce* değeri içerir ve bu *nonce* değeri doğrulayıcı düğümler tarafından sık sık değiştirilerek bloğun başlığının farklı özet değerini elde etmektedir. Hesaplanan özet değerinin, belirli bir değere eşit veya daha küçük olması gerektirir. Bir düğüm hedef değeri elde ettiğinde, yeni bloğu diğer düğümlere yayınlar; özet değerinin doğruluğunu, tüm diğer düğümlerin karşılıklı olarak onaylamaları gerekmektedir. Yeni blok, diğer doğrulayıcı düğümler tarafından doğrulandığı takdirde,

Blockchain defterine eklenmektedir. Proof-of-Work hesaplama adımları şu şekilde ifade edilmektedir [70]:

(1) Zorluk derecesi elde edilmesi: Her 2016 tane bloğunun oluşturulmasından sonra, zorluk derecesi dinamik olarak ayarlanır.
(2) Kayıtların toplanması: Son bloğun üretilmesinden sonra bekleyen tüm kayıtlar, bir blok haline getirilmek amacıyla toplanır.
(3) Hesaplama: 0 ile $2_{32}$ aralığından bir *nonce* değeri seçilir ve çift hash-256 değerini hesaplanır. Özet değeri, hedef değerden küçük veya eşit olursa, blok yayınlanacaktır.
(4) Yeniden başlatılması: Belirli bir zamanda özet değeri hesaplanamıyorsa, ikinci adıma dönülmektedir. Bir düğüm hesaplamayı tamamlarsa, Proof-of-Work hesaplaması birinci adımdan yeniden başlar.

Asimetrik bir yapısı olan Proof-of-Work algoritması, çözülmesi zor ve doğrulaması çok kolaydır [71]. Özellikle bir madenci düğümü, özet bulmacayı çözümü bulmak için çok zamana ihtiyaç duyar, ancak ağdaki diğer madenciler bulunan çözümün geçerliliğini derhal doğrulayabilir. Şekil 2.6'te, bir PoW uygulama örneği gösterilmektedir. Zincire bir blok eklemek için bir madenci düğümü, nonce ile önceki blok özeti değerlerinin özetini, belirli bir eşikten (threshold) daha düşük bir sonuç değeri vereceği şekilde bulmalıdır. Bu genellikle belirli sayıda, ardışık sıfır (0) ile başlayan bir değerdir. Ardışık sıfırların sayısı, bulmacayı çözmenin zorluğuyla ilgilidir ve Blockchain ağı tarafından dinamik olarak ayarlanır. Madenci düğümler, eşik değerden daha düşük bir sonuç elde edene kadar rasgele özetleme fonksiyon uygular ve bu durum, ağ tarafından yeni blokların üretilme hızını sınırlamaktadır.




**Şekil 2-6:** PoW hesaplama mantığı.

Proof-of-Work algoritmasında, çözülmesi gereken bulmaca, SHA-256 özet bilgisine dayanmaktadır [72]. Özet fonksiyonlarının en önemli özelliği, farklı uzunluklardaki verilerin belirli bir uzunluğa dönüştürmesi ve aynı girdi ile her zaman aynı çıktıyı vermesidir; böylece verilerde oluşan en küçük değişiklik, özet sonucunu tamamen değişmesine neden olacaktır.

Proof-of-Work algoritmasında doğrulayıcı düğümler, Blockchain'e eklenen en son bloğun dışında kalan kayıtları bir *aday bloğu* haline getirir [41]. Aday bloğunun Blockchain ağına eklenmesi için, belirli bir hedef değerinden daha küçük bir özet değeri bulunması gerekir. Bu özet değerin hesaplanması için, aday bloğunun başlığındaki bileşenleri sabitleyip *nonce* değeri değiştirilir. Şekil 2.7'de gösterildiği gibi, 'Blockchain' kelimesi aday bloğun sabit bileşenleri yerine geçer ve *nonce* ise, onun arkasına eklenen sayıdır. Bu hesaplamanın zorluluğunu arttırmak için hedef değerinin 0 ile başlaması gerekmektedir. Tek sıfırla başlayan bir özet değeri bulmanın olasılığı 1/16'dır; hedef değerin başındaki sıfır sayısı arttıkça, *nonce* değerin bulunması daha da zorlaşmaktadır [73]. *Nonce* değerini bulan ilk doğrulayıcı, çözümün geçerliliğinin kontrol edilmesi için diğer doğrulayıcılar ile paylaşır. Çözümün geçerliliğini kontrol edilmesi için, tek bir hesaplama işi gerekecektir.


**Şekil 2-7:** SHA-256 özet hesaplama örneği.

### 2.2.2. Blockchain Sistemleri

Mevcut Blockchain sistemleri Tablo 3'te gösterildiği üzere üç gruba ayrılmaktadır: Açık (Public) Blockchain, Konsorsiyum (Consortium) Blockchain ve Özel (Private) Blockchain [74].

Açık Blockchain sistemlerinde, tüm kayıtlar halka açıktır. Bu ağda, tüm düğümler uzlaşma sürecine katılabilir ve merkezi olmayan bir ağ türüdür [75]. Konsorsiyum Blockchain sistemlerinde, sadece bir grup önceden seçilmiş düğüm uzlaşma sürecine katılabilir [76]. Birden çok organizasyon tarafından inşa edilen Konsorsiyum Blockchain ağında, uzlaşmaya varılması için birkaç düğüm seçilir ve dolayısıyla "kısmen" merkezi olmayan bir ağ türüdür [77]. Özel Blockchain sistemlerinde, yalnızca belirli bir organizasyondan gelen düğümlerin uzlaşma sürecine katılmasına izin verilmektedir. Bir Özel Blockchain sistemi, tek bir organizasyon tarafından kontrol edildiği için, merkezi bir ağ olarak kabul edilmektedir [77].

**Tablo 2.4:** Blockchain sistemleri.

| Özellik | Açık Blockchain | Konsorsiyum Blockchain | Özel Blockchain |
|---|---|---|---|
| Konsensüs tayini | Tüm düğümler | Seçilmiş düğümler kümesi | Tek bir düğüm |
| Okuma izni | Açık | Açık ve Kısıtlı | Açık veya Kısıtlı |
| Verimlilik | Düşük | Yüksek | Yüksek |
| Merkezi | Hayır | Kısmen | Evet |
| Konsensüs Süreci | İzinsiz | İzinli | İzinli |

### 2.2.3. Blockchain Platformları

Günümüzde Ethereum [78], Hyperledger [79], Ripple [80], Quorum [81], Corda [82] gibi birçok farklı Blockchain platformları bulunmaktadır. En yaygın kullanılan Blockchain platformları, Ethereum ve Hyperledger platformlarıdır.

Ethereum, akıllı sözleşme (smart contract) işlevselliğine sahip açık kaynak kodlu ve halka açık Blockchain tabanlı dağıtık bir bilgisayar platformudur. Bitcoin uygulamalarını genişletmeye imkân sağlar. Sahip olduğu Ethereum Sanal Makine (Ethereum Virtual Machine) teknolojisi ile, özel iş modellerinin geliştirilmesini, yani akıllı sözleşmelerin farklı uygulamalarda kullanılmasını sağlar [83]. Ethereum platformunun açık kaynak kodlu olması, yazılımın geliştiriciler tarafından indirilerek özel bir ağ olacak şekilde yapılandırılmasını mümkün kılmaktadır.

IBM tarafından sunulan Hyperledger Fabric platformu, farklı endüstriler için Blockchain uygulamaları geliştirmeyi ve özel -yani izinli- Blockchain sistemi oluşturmayı sağlayan bir platformdur. Hyperledger mimarisi modüler bir yapıdadır; modüller (örneğin uzlaşma hizmeti, üyelik hizmeti ve benzeri bileşenler) tak-ve-çalıştır (plug-and-play) özelliğine sahiptir [84]. Hyperledger platformunun çalışma mantığı, "chaincode" olarak adlandırılan akıllı sözleşmelere

ve akıllı sözleşmelerin çalışmasına olanak sağlayan konteyner (docker) teknolojisine dayanır [85].

Blockchain platformlarının analizi ve karşılaştırması konularında çok sayıda akademik çalışmalar yapılmıştır. Pongnumkul ve diğerleri [86], özel bir Ethereum Blockchain ağı ve özel bir Hyperledger Blockchain ağı üzerinde farklı sayıda kayıtlar ile, performans analizini gerçekleştirmiştir. Analiz sonucunda, Hyperledger platformunun Ethereum platformuna kıyasla daha yüksek verimlilik ve az gecikme sağladığı görülmüştür. Ayrıca, bu iki platform arasındaki verimlilik farkı, işlem sayısı arttıkça daha da belirginleştiği görülmüştür. Bununla birlikte, Ethereum platformunun benzer hesaplama kaynakları için çok fazla sayıda eşzamanlı olarak kayıtları gerçekleştirebildiği görülmüştür.

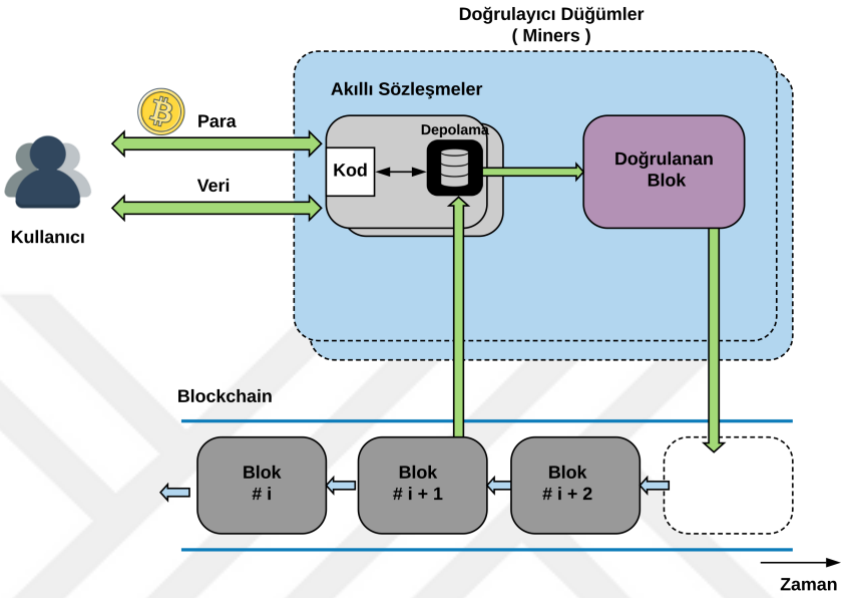
Bir diğer çalışmada [87] ise, Ethereum ve Hyperledger platformlarının çeşitli açılardan oldukça esnek olduğunu belirtmiştir. Bununla beraber, Ethereum platformunun sağladığı güçlü akıllı sözleşme altyapısı, izinsiz çalışma şekli, şeffaflığı, performans ölçeklenebilirliği özellikleri ile, her tür uygulama için genel bir platform haline geldiği belirtilmiştir. İzinli çalışma şekline sahip Hyperledger platformunun ise, özellikle Byzantine Fault-Tolerant (BFT) uzlaşma algoritması ve güçlü erişim kontrolü özelliklerini kullanarak performans ölçeklenebilirliği ve gizlilik sorunlarını çözdüğü vurgulanmıştır.

### 2.2.4. Akıllı Sözleşmeler

Blockchain sistemleri, hesaplama sonuçlarını kaydeden defter ile birlikte çalışabilen genel amaçlı programlanabilir bir altyapı sağlamaktadır. Blockchain üzerinde depolanan ve çalıştırılabilen programlar akıllı sözleşmeler (smart contracts) olarak bilinir [88]. Akıllı sözleşmeler, daha karmaşık programlanabilir işlemler için kullanılır; tetikleyicileri, koşulları veya çalışma/iş mantığını [89] ifade etmektir.

Ethereum tabanlı bir akıllı sözleşme, bilgiyi depolayan, girişleri işleyen, çıkışları kaydeden ve -önceden tanımlanmış- belirli koşullar yerine getirildiği takdirde dışarıdan erişilebilen bir şifreleme kutusudur [74]. Ethereum, akıllı sözleşmelerin kolay uygulanmasını sağlar [49]. Akıllı sözleşmenin bir fonksiyonu çağrıldığında, Ethereum Network tarafından ilgili kod çalıştırılır.

Şekil 2.8'te gösterildiği üzere, akıllı sözleşmeler düğümler tarafından geliştirilmektedir. Akıllı sözleşmenin durum bilgisi, Blockchain üzerinde saklanır. Kullanıcılar akıllı sözleşmelere para (transfer işlemi) veya veri gönderebilir; aynı zamanda akıllı sözleşmelerden kullanıcılara para ve veri gönderilmektedir.




**Şekil 2-8:** Akıllı sözleşmeler ile sistem bileşenleri ve etkileşimi.

Blockchain Kapsamında Akıllı Ortamlardaki Güvenlik Zorlukları IoT'nin içsel güvenlik önlemlerinin zayıflığı nedeniyle, IoT sistemlerini güvenlik ve gizlilik tehditlerine karşı savunmasız bırakmaktadır [90]. Minhaj Ahmad Khan ve Khaled Salah'ın gerçekleştirdiği bir çalışmada [91] bahsedildiği üzere, temelinde güvenlik sağlama amacına sahip olan Blockchain teknolojisi, IoT sistemlerindeki büyük güvenlik gereksinimlerinin karşılama potansiyeline sahiptir. Değişmezlik (immutability), şeffaflık (transparency), denetlenebilirlik (auditability), veri şifreleme ve benzeri yetenekleri ile Blockchain, IoT'nin çoğu mimari eksikliğinin çözülmesine yardımcı olabilmektedir.

IoT sistemleri için en önemli güvenlik zorluklarından, dağıtık mimari yapısına sahip olmasıdır [92]. Bir IoT ağında, genellikle her bir düğüm Hizmet Dışı Bırakma (Distributed Denial of Service, DDoS) ve benzeri siber saldırıları başlatmak için kullanılabilir [93]. Saldırıya maruz kalmış bir cihaz içeren düğüm, ilgili IoT sistemini hızla çökertebilir. Diğer bir önemli bir konuda, merkezi bir buluta dayalı yapılandırmalara ilişkindir [94]. Aynı zamanda, IoT

sistemleri için güvenlik, verilerin gizliliği ve bütünlüğü konularında verimli ve etkin çözümlere ihtiyaç duyulmaktadır; güvenli kimlik doğrulama ve erişim kontrol (access control) mekanizmalarının sağlamasını gerekmektedir [95].

Son yıllarda, araştırmacılar Blockchain teknolojisinin IoT sistemlerine bütünleştirilmesi ve ilgili sorunlarını ele almaktadırlar [96, 97]. Blockchain teknolojisinin, IoT sistemlerinin karşılaştığı problemleri, özellikle kimlik doğrulama ve erişim kontrolü problemlerini çözme potansiyeline sahip olduğu görülmektedir. Atzori ve diğerleri [98], IoT sistemleri için Blockchain tabanlı platformları incelemiştir; Blockchain teknolojisini IoT ortamında uygulamanın sınırlarını vurgulamıştır. Reyna ve diğerleri [99], IoT ve Blockchain teknolojisinin entegrasyonundan kaynaklanan zorlukları analiz etmişlerdir; bu kapsamda IoT ve Blockchain teknolojisini bütünleştiren olası entegrasyon yolları ve platformları sunulmuştur. Jesus ve diğerleri [100], IoT sistemlerini güvence altına almak için Blockchain teknolojisinin uygulamasını araştırmıştır ve "Stalker" saldırısını incelemiştir. Diğer bir çalışmada [48], Blockchain teknolojisinin IoT'deki büyük güvenlik gereksinimlerinin ele alınmasına yardımcı olacağı vurgulanmıştır. Blockchain teknolojisinin sahip olduğu değişmezlik, şeffaflık, denetlenebilirlik, veri şifreleme ve operasyonel esneklik gibi yeteneklerin, IoT'nin mimari eksikliklerinin çoğunun çözümlenmesine yardımcı olacağı belirtilmiştir.

Blockchain teknolojisi erişim kontrol mekanizmaları açısından da önem kazanmaktadır [101]. Bilgisayar sistemlerinde güvenliği sağlamak için temel bir mekanizma olan erişim kontrolü, bazı güvenlik modelleri ve politikalarına göre hangi nesnelerin hangi iletişim haklarına sahip olacağına karar veren bir süreçtir [102]. Etkili bir erişim kontrol sistemi ile, gizlilik, bütünlük ve kullanılabilirlik gibi temel güvenlik gereksinimlerini karşılanabilir. Blockchain tabanlı bir iletişim altyapısında dört evrensel güvenlik ilkesi sağlanması mümkün olacaktır:

(1) Kimlik Doğrulama: Kişisel olarak Blockchain ağına katılmak isteyen kullanıcıların gerçekleşen iletişimde kimliğinin onaylanması ve doğrulanması sağlanacaktır.
(2) İnkâr Edememe: Bir aktör gerçekleştireceği bir işlemden sonra söz konusu gerçekleştirdiği işlemi inkâr edemeyecektir.
(3) Gizlilik: Sistemde veriler sadece yetkili aktörler tarafından erişilebilir olacaktır ve yetkisi olmayan kişilerin eline geçmesi engellenecektir.
(4) Veri Bütünlüğü: Veri transferi esnasında veri bütünlüğünün korunması esastır ve değiştirildiğinde farkına varılacaktır.

Blockchain teknolojisinin IoT bağlamında kimlik doğrulama ve erişim kontrolü güvenlik gereksinimlerini karşılamak için umut verici bir teknoloji olduğu vurgulanmaktadır [103-105]. Blockchain tabanlı kimlik doğrulama ve erişim kontrol konusundaki mevcut çalışmalar (state-of-the-art) şu şekilde özetlenebilir:

- Yapılan bir çalışmada [106], Ethereum platformu üzerinde IoT cihazların Blockchain tabanlı tanımlanması ve kimlik doğrulaması modeli önerilmektedir. Önerilen "Bubbles of Trust" modeli, akıllı sözleşmeleri uygulayan halka açık Blockchain ağı kullanarak ayrı güvenli sanal bölgeler oluşturma fikri üzerine kurulmuştur. Oluşturulan her sanal bölge için bir yönetmen (master) seçilir. Bir sanal bölgeye ait cihazların, kendilerini kayıt ettirmeleri gerekmektedir. Bölgeye katılan her cihaz için yönetmen tarafından hazırlanıp imzalanan bir bilet (ticket) oluşturulur. Sistemin güvenliğini sağlamak amacıyla, iletişim sadece aynı sanal bölgede kayıtlı olan cihazlar arasında gerçekleşir. Bir sanal bölge dışında kalan bir cihazdan gelen bir talep, tehdit içeren bir talep olarak kabul edilmektedir.
- Xia ve diğerleri [107], Blockchain teknolojisinin değiştirilemez ve yerleşik özerklik (built-in autonomy) özelliklerini kullanarak, bulut ortamında depolanan hassas medikal verilerle ilgili erişim kontrolü zorluklarını karşılayan bir Blockchain tabanlı veri paylaşım modeli önermektedir. Önerilen model, sadece izin verilen ve doğrulanmış kullanıcılara erişim sağlayan bir Blockchain altyapısına dayanmaktadır. Özel (izinli) Blockchain ağı ve güvenli şifreleme tekniklerini kullanarak erişim kontrol modeli sağlanmaktadır. Kullanıcıların kimliklerini ve şifreleme anahtarlarını doğruladıktan sonra, model üzerinden medikal verilere erişmelerine izin verilmektedir.
- Diğer önemli bir çalışmada [108], IoT cihazlarına kullanıcı erişimini yönetmek amacıyla, Ethereum akıllı sözleşmelerine bağlanabilen Blockchain ağı ve etkin sis (fog) düğümlerini kullanan, merkezi olmayan ve ölçeklenebilir bir kimlik doğrulama mekanizması önerilmektedir. Erişim kontrolü, güvenilir üçüncü tarafa ihtiyaç duymadan akıllı sözleşmeler tarafından gerçekleştirilmektedir.

# 3. MALZEME VE YÖNTEM

Tezin amacı, akıllı ortamlarda birbirine güvenmeyen aktörler arasında Blockchain tabanlı güvenli bir model altyapısını hazırlayarak aktörler arasında hızlı ve güvenli mutabakat sağlanması, modele katılacak yeni kullanıcılara kimlik doğrulaması modeli ve erişim kontrol mekanizması sağlamaktır.

Tez dokümanının bu bölümünde, tez çalışmasında uygulanan yöntem, sistem analizi çalışmaları ve Blockchain tabanlı kimlik doğrulama mimarisinin ve erişim kontrol mekanizmasının sağlaması gereken iletişim ve güvenlik gereksinimleri detaylandırılmıştır.

## 3.1. TEZ ÇALIŞMASININ YÖNTEMİ

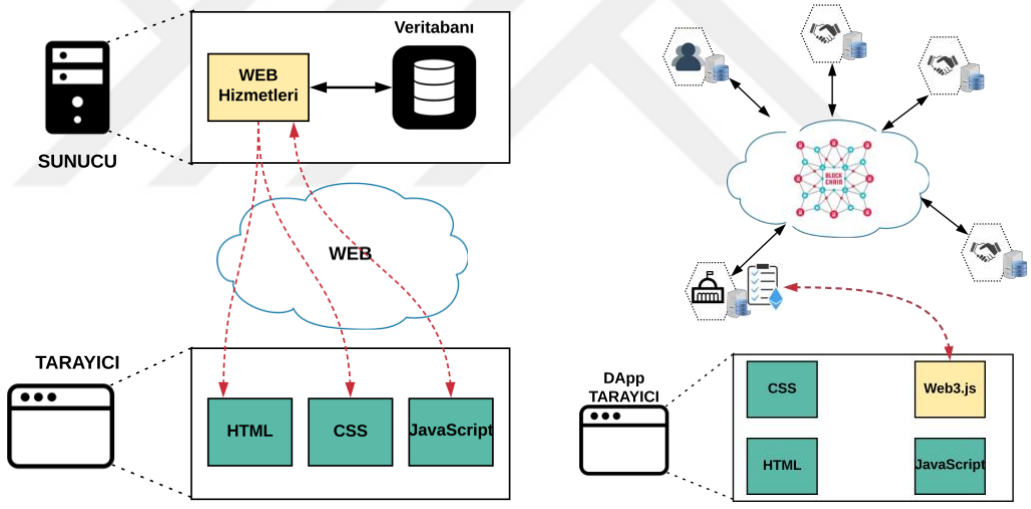
Geliştirilecek akıllı ortamlarda Blockchain tabanlı kimlik doğrulama sistemine temel teşkil eden konular literatürde incelenmiştir. Bu doğrultuda, yapılan akademik ve teknik ön araştırmalar sonucunda geliştirilecek olan modelin, *Ethereum Blockchain platformunu* ve *Ethereum akıllı sözleşmelerin* kullanılmasına karar verilmiştir.

Araştırma tasarımının ilk aşamasında sistem gereksinimlerinin, bileşenlerinin ve mimarisinin analizi gerçekleştirilmiştir. Sistem analizi çalışmalarına göre, önerilen Ethereum Blockchain tabanlı kimlik doğrulama sistemi üç temel bileşenden oluşmaktadır ve sistem geliştirme çalışmaları bu kapsamda gerçekleştirilmiştir: (a) Ethereum Blockchain Ağı, (b) Akıllı Sözleşmeler, (c) Dağıtık Uygulamalar.

**Ethereum Blockchain Ağı:** Akıllı ortamlarda Blockchain tabanlı kimlik doğrulama modelinin geliştirilmesi amacıyla, *Geth* aracı ile Blockchain ağı oluşturulmuştur. Farklı servis sağlayıcı aralarında kullanılacak olan bir konsorsiyum Blockchain ağı geliştirilmiştir.

**Akıllı Sözleşmeler:** Blockchain bağlamında, akıllı sözleşme kendi kendini çalıştırabilen ve gerekli olan koşullar sağlandığında bir işlemi gerçekleştirebilen bir bilgisayar programıdır. Önerilen modelde son kullanıcıların işlemlerini kaydetmek ve belirli servis sağlayıcının verilerine kim tarafından erişebileceği konusunda erişim kontrol mekanizması sağlayabilmesi için kullanılmıştır. Akıllı sözleşmeleri geliştirmek için, *Solidity* kodlama dili ile *Truffle* ortamı kullanılmıştır.

**Dağıtık Uygulama (Distributed Application, DApp):** Merkezi olmayan uygulama için kullanılan bir kısaltmadır. Dağıtık uygulamalar iki kısımdan oluşmaktadır: (1) Herhangi bir yazılım dili ile yazılabilen Önyüz (Frontend), ve (2) Akıllı sözleşmelerden oluşan ve Blockchain üzerinde çalışan Sunucu Uygulaması (Backend). Şekil 3.4'te DApps ile geleneksel uygulama arasındaki fark kısaca gösterilmektedir. Geleneksel uygulamalarda web üzerinden merkezi olan sunucuya hep iletişim kuracak iken, dağıtık uygulamaların temelinde herhangi bir düğüme bağlanabilme özelliği vardır. DApp, önerilen kimlik doğrulama modelindeki aktörlerin, Blockchain ağı ve bu ağa yüklenen akıllı sözleşme ile iletişim kurmasını sağlamaktadır. Önyüz (frontend) ve Sunucu tarafı (backend) kısımları için, *Visual Studio Code* aracı ile HTML5 ve JavaScript dili kullanılmıştır; *lite-server* sunucu uygulaması ile sistemin çalışması sağlanmıştır.




(a) Geleneksel yapısı. (b) Dağıtık yapısı.

**Şekil 3-1:** Geleneksel ve dağıtık yapı uygulamaları.

## 3.2. SİSTEM ANALİZİ

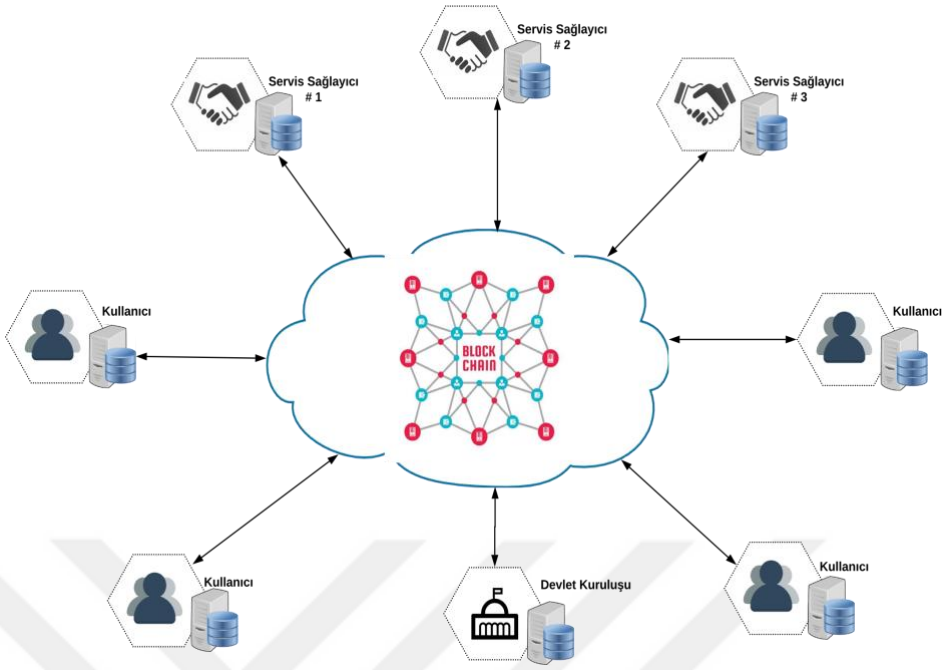
### 3.2.1. Sistem Aktörleri ve Mimarisi

Geliştirilecek olan Blockchain tabanlı kimlik doğrulama sistemi Şekil 3.1'de gösterildiği üzere üç ana aktörden oluşmaktadır: (a) Servis Sağlayıcı (Service Provider), (b) Son Kullanıcılar (End Users), (c) Devlet Kuruluşu.

**Servis Sağlayıcı:** Çeşitli katma değerli servisler sunan, yöneten ve kontrol eden aktördür. Sistemin kullanılabilmesi için öncelikle servis sağlayıcılar aralarında bir konsorsiyum Blockchain ağı oluşturulmaktadır.

**Devlet Kuruluşu:** Oluşturulan Blockchain ağının güvenilir üçüncü tarafını temsil eden aktördür. Devlet Kuruluşu düğümünün temel amacı, Blockchain ağına katılmak isteyen son kullanıcılarının kimliğini doğrulamasıdır.

**Son Kullanıcılar:** Blockchain ağı dışında bulunan aktörlerdir. Son kullanıcı, Blockchain ağına katılarak, belirli servis sağlayıcı veya servis sağlayıcılar tarafından sunulan servisi veya servisleri kullanacaktır. Blockchain üzerinde yapılan her işlem, ağdaki aktörler tarafından görüntülenmektedir. Son kullanıcıların Konsorsiyum Blockchain ağına katılmadan önce kimlik doğrulama sürecinden geçmesi gerekmektedir. Kimliği doğrulanmış son kullanıcılar, servis sağlayıcılar tarafından gerçekleştirilen işlemleri sorgulayabilecek ve erişebilecektir.




**Şekil 3-2:** Sistem aktörleri.

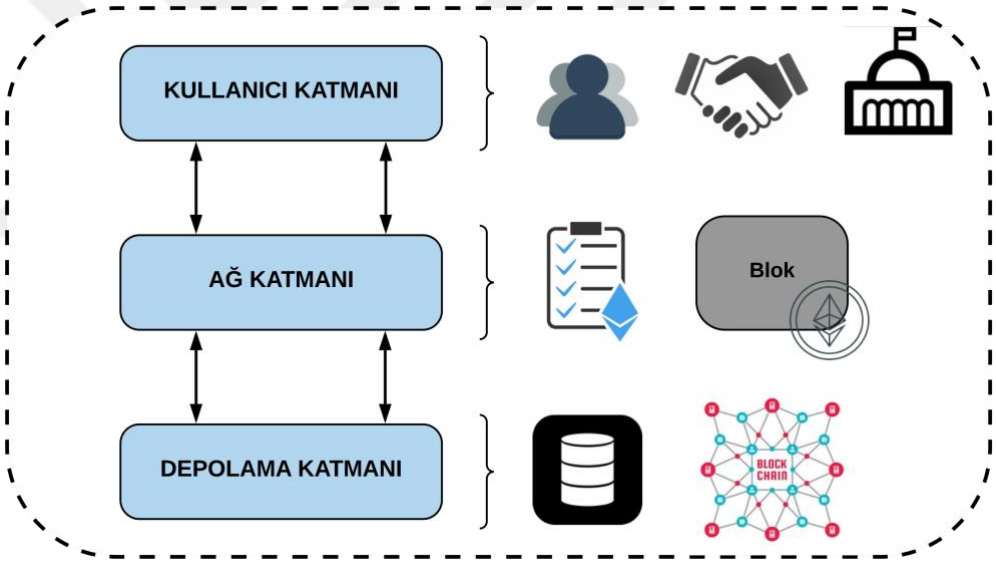
Önerilen modelin sistem mimarisi Şekil 3.2'de gösterildiği üzere üç katmanlı bir yapıdan oluşmaktadır: (a) Kullanıcı Katmanı, (b) Ağ Katmanı ve (c) Depolama Katmanıdır.

**Kullanıcı katmanı:** Modeldeki tüm aktörleri -servis sağlayıcılar, son kullanıcılar, devlet kuruluşu- kapsayan katmandır. Her bir aktör, bir Blockchain düğümü olarak kabul edilmektedir. Blockchain ağına katılan aktörlerin her biri için bir açık ve özel anahtar çifti üretilir. Aktörler tarafından yapılan her işlem, *web3js (JavaScript library)* kütüphanesi üzerinden Blockchain ağına yeni bir işlem (kayıt) olarak gönderilecektir. Bu kayıt, ilgili aktörün özel anahtarı ile imzalanarak Blockchain ağına gönderilecektir ve ağa eklenmeden önce doğrulayıcı düğümler tarafından doğruluğu kontrol edilecektir. Doğrulayıcı düğüm tarafından onaylanan kayıt, bir daha değiştirilmemek üzere Blockchain ağına eklenecektir ve aynı anda ilgili diğer aktörler tarafından görüntülenebilecektir. Böylece, tüm aktörler arasında ilgili veri üzerinde mutabakat sağlanacaktır. Aktörün özel anahtarı ile kayıt bilgisinin imzalanması da, reddedilemezlik güvenlik gereksinimi sağlayacaktır.

**Ağ katmanı:** Kullanıcı katmanındaki aktörler arasında güvenli bir veri yönetim imkânı sağlanacaktır. Her düğüme ait özel bir adres oluşturulur. Aktörler *web3js (JavaScript library)*

kütüphanesi kullanarak ağ katmanı ile iletişim kurabilecektir. Bu katman Ethereum Blockchain ağı ile Akıllı Sözleşmeleri içermektedir. Ethereum Blockchain olarak Konsorsiyum Blockchain ağı oluşturulacaktır. Bu ağ türü, önceden seçilmiş bazı düğümler (servis sağlayıcılar) tarafından kontrol edilen kısmen halka açık olan ve kısmen izin verilen bir sistem olacaktır. Akıllı Sözleşmeler kapsamında ise, taraflar arasındaki veri iletişimini yönetmek amacıyla, kendi kendisini yürütebilen, kodlardan oluşan bir program kullanılacaktır. Sözleşmede yer alan kod ve anlaşma unsurları, dağıtık Blockchain ağında bulunacaktır.

**Depolama katmanı:** Geliştirilecek modelin son katmanıdır. Bu katmanda bir aktörün kullandığı veri tabanı sunucusu bulunmaktadır. Sistemde yapılan tüm kayıtlar, Blockchain defterinde (ledger) tutulacaktır. Ayrıca düğümler, sistem dışında oluşan bilgileri tutmak için kendilerine ait veri tabanı sunucusunu kullanabilirler.




**Şekil 3-3:** Model mimarisi.

### 3.2.2. Sistem Güvenlik Gereksinimleri ve Gerçekleşme Aşamaları

Geliştirilecek olan modelin iki temel güvenlik unsuru bulunmaktadır: (1) Kimlik Doğrulaması ve (2) Erişim Kontrol Yönetimi.

(1) Bir sistemin güvenliğinin sağlanması için öncelikli adımlardan biri; kullanıcının sisteme kayıt aşamasında kimlik doğrulama (user authentication) işleminin gerçekleştirilmesidir.

Kimlik doğrulaması, çeşitli güvenlik araçları ve yöntemleri kullanarak kullanıcının kim olduğunu kanıtlama işlemidir.

(2) Sistem güvenliğinin ikinci önemli adımı; Blockchain ağında bulunan (kimliği kayıt aşamasında doğrulanmış) düğümlerin sakladıkları verilere erişim kontrol (access control) mekanizmasının belirlenmesidir. Erişim kontrol mekanizması, verilerin kimin tarafından ve hangi koşullarda erişilebileceğinin yönetilmesi işlemidir.

Önerilen modelde kullanılacak olan Konsorsiyum Blockchain ağında yer alan son kullanıcılar, Blockchain ağına katılma aşamasında yani sisteme kayıt olma (system registration) aşamasında bir kimlik doğrulama sürecinden geçmesi gerekmektedir. Bahse konu kimlik doğrulama süreci iki aşama içermektedir:

(1) İlk aşamada, kullanıcı hizmet almak istediği servis sağlayıcısı düğümüne telefon numarası ile birlikte istenilen bazı kişisel bilgileri gönderir. Servis sağlayıcısı düğümü, kullanıcının sağladığı telefon numarasına tek kullanımlık şifre (One Time Password, OTP) bilgisi göndererek, kullanıcının iddia ettiği kişi olduğunu kontrol eder.

(2) İkinci aşamada, ilgili servis sağlayıcısı düğümü ilgili kullanıcının sağlamış olduğu bilgileri bir kayıt haline getirir ve özel anahtarıyla imzalar. Ardından, kimlik doğrulama sürecini tamamlamak üzere -güvenilir bir üçüncü taraf olarak- devlet kuruluşu düğümüne gönderir. Devlet kuruluşu düğümü, kendisine gelen kullanıcı bilgilerini, sunucularında bulunan kullanıcı bilgileri ile kontrol eder.

Şekil 3.3'te gösterildiği üzere, sisteme kayıt aşamasında uygulanacak olan kimlik doğrulama süreci sırasıyla şu adımları içermektedir:

(1) Son kullanıcı, belirli bir servis sağlayıcısına belirli bir servisi kullanmak istediğini -mobil veya web ortamı üzerinden- talep göndermek suretiyle beyan eder. Servis sağlayıcı kullanıcıdan telefon numarası ile birlikte bazı kişisel bilgileri ister. Son kullanıcı, istenilen bilgileri ilgili servis sağlayıcısına gönderir.

(2) Servis sağlayıcı, kullanıcının gerçek biri olduğu ve iddia ettiği kişi olduğunu tespit etmek amacıyla, aldığı telefon numarası bilgisine OTP gönderir.

(3) Kullanıcının doğru OTP bilgisi sağlaması durumunda, servis sağlayıcısı düğümü kullanıcının sağladığı bilgileri kayıt haline getirerek kendi özel anahtarı ile imzalar.

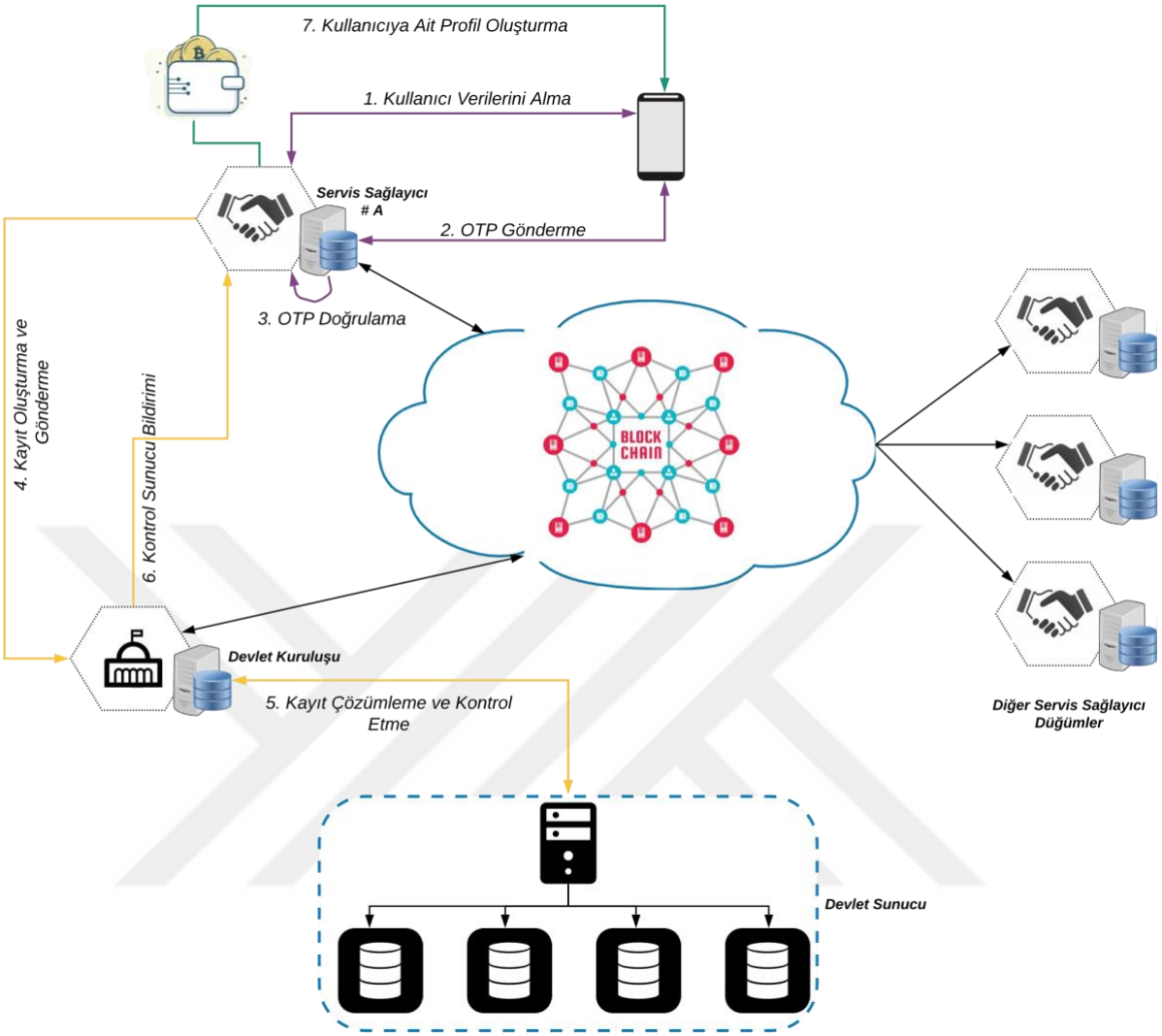
Kullanıcının yanlış OTP bilgisi sağlaması durumunda, kullanıcıya işlem başarısızlığı bildirimi gönderilir.

(4) Servis sağlayıcısı düğümü, Blockchain ağında bulunan -güvenilir bir üçüncü taraf olarak- devlet kuruluşu düğümüne imzalanmış kaydı gönderir.

(5) Devlet kuruluşu düğümü gönderilen kaydı, servis sağlayıcısı düğümünün açık anahtarı ile çözerek, kayıt içinde sağlanan kullanıcı kimlik bilgilerini, kendi sunucusunda bulunan bilgiler ile uyumluluğunu kontrol eder.

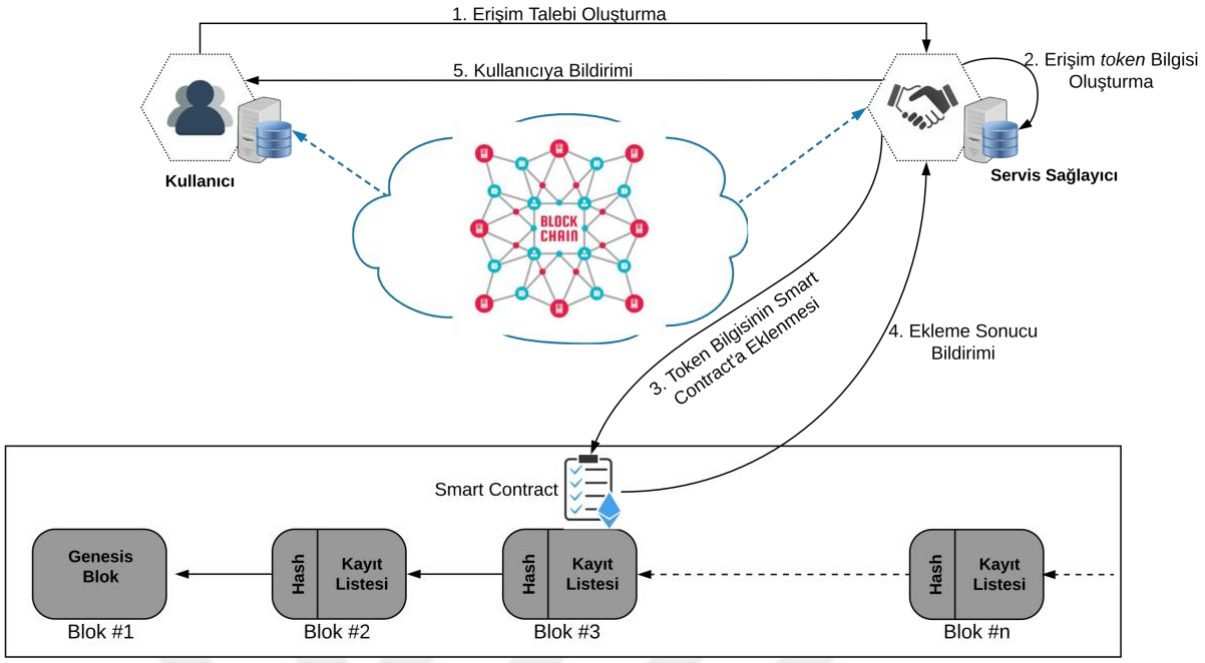
(6) Devlet kuruluşu sunucuları, kontrol sonucunu -doğru veya yanlış şekilde- devlet kuruluşu düğümün geri gönderir ve sonuç devlet kuruluşu düğümü tarafından servis sağlayıcısı düğümüne iletilir.

(7) Son kullanıcının kimlik doğrulama sonucu doğru ise, servis sağlayıcısı düğümü, son kullanıcıya bir profil oluşturma işlemini gerçekleştirir. Son kullanıcıya ait bir çift anahtar (özel ve açık anahtar) bilgisini ile rastgele oluşturulan bir PIN değeri bilgisini kullanıcının telefonuna gönderir. PIN bilgisi, kullanıcının sonraki kullanımlarında sisteme giriş yapabilmesi amacıyla kullanılacaktır. Son kullanıcının kimlik doğrulama sonucu yanlış ise, kullanıcıya işlem başarısızlığı bildirimi gönderilir.




**Şekil 3-4:** Kimlik doğrulama süreci akışı.

Önerilen modelin Konsorsiyum Blockchain ağında yer alan son kullanıcıların sistem kullanımı aşamasında -örneğin Blockchain ağında bulunan bir Servis sağlayıcısı düğümünden bir bilgi sorgulaması veya bilgi görüntülemesi durumunda- ise, kullanıcının ilgili veriyi kullanabilmesi için bir erişimi kontrol sürecinden geçmesi gerekmektedir. Erişim kontrol süreci için öncelikli adım, sistemde kayıtlı ve aktif olan her bir servis için, ilgili servis sağlayıcılarının gerekli yetkilendirme bilgisini -token bilgisini- ilgili kullanıcı(lar) için tanımlaması gerekmektedir; bu bilgiler Akıllı Sözleşmelerde kayıt altına alınacaktır. Bir servisi kullanmak için uygun ve geçerli *token* bilgisine sahip olan kullanıcı, talep ettiği ilgili servisi kullanabilecektir.

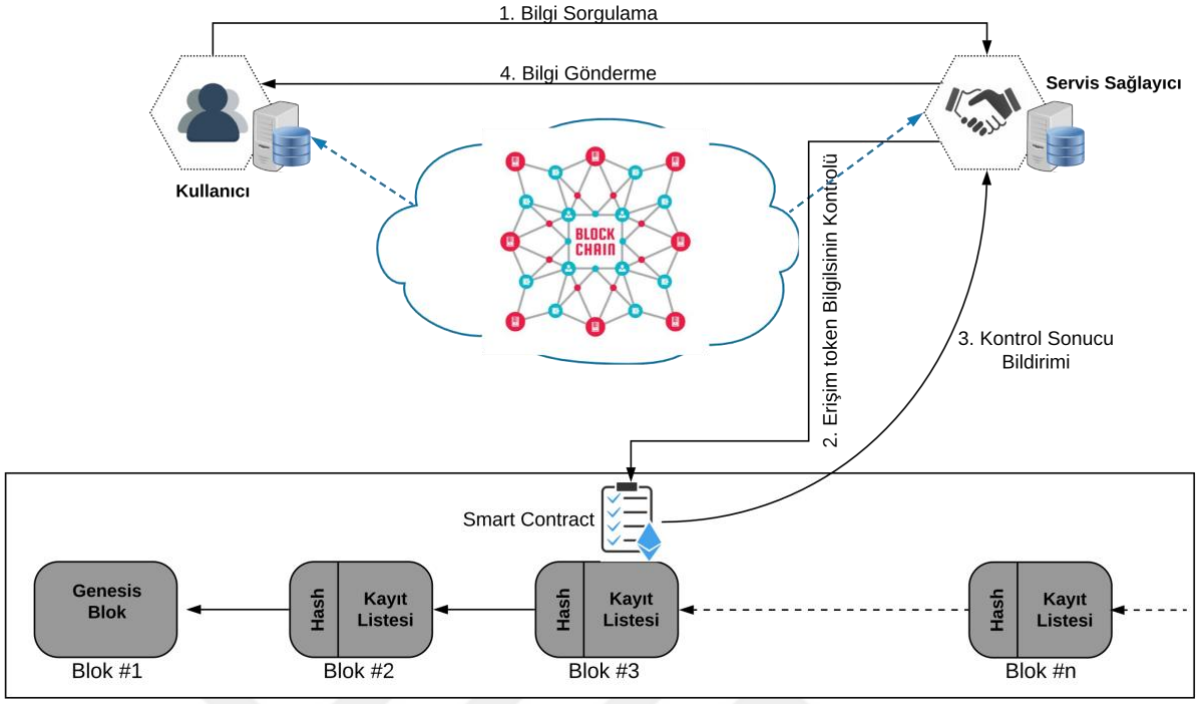



**Şekil 3-5:** Erişim token bilgisini oluşturma süreci.

Şekil 3-4'te gösterildiği üzere erişim kontrol süreci, kullanıcının erişmek istediği servisi, ilgili servis sağlayıcı düğümüne talep göndermek ile başlamaktadır. Servis Sağlayıcının, kullanıcı için erişim token'i oluşturup akıllı sözleşme'ye eklmesi gerekmektedir. Token bilgisi oluşturulması için ilk önce üç tane bilgi-erişmek istenilen servis numarası, rastgele sayı ve token geçerlilik süresi- birleştirilir. Ardından elde edilen değer SHA-3 fonksiyonunu kullanarak özet bilgisini hasaplanır. Özet değeri akıllı sözleşmeye eklenmeden önce kullanıcının özel anahtar ile imzalanacaktır.

```
createToken: function (serviceNo, randomVal, tokenPeriod) {
    var record = serviceNo + randomVal + tokenPeriod;
    var hashedRecord = web3.sha3(record);
    var signedRecordWithUserKey = web3.eth.sign(userKey,
hashedRecord).slice(2);
    App.contracts.Profiles.deployed().then(function (instance) {
      instance.addAccessToken(userKey, signedRecordWithUserKey, {
        from: App.allAccounts[0],
        gas: 500000
      }).then(function (token) {
          alert("Erisim Token basari ile olusturuldu: " + token)
      }).catch(function (err) {
        console.error(err);
      });
    });
```

**Şekil 3-6:** Erişim token bilgisini oluşturma fonksiyonu.




**Şekil 3-7:** Erişim kontrol süreci.

Şekil 3-6'te gösterildiği üzere kimlik doğrulaması tamamlanmış ve sisteme kayıtlı olup erişim token bilgisine sahip olan bir kullanıcının erişim kontrol süreci sırasıyla şu adımları içermektedir:

(1) Kimlik doğrulaması tamamlanmış ve sisteme kayıtlı olan bir kullanıcı, web uygulama üzerinden kullanmak istediği servisi seçerek servis kullanma talebini Blockchain ağına gönderir. Kullanıcı, web uygulama üzerinden belirli servis ile ilgili bilgi alma talebini servis sağlayıcıya gönderir. Gönderilen talebin içeriğinde kullanıcının adresi, servis numarası, servis sağlayıcının adresi ve token bilgisi bulunmaktadır.

(2) Akıllı Sözleşme tarafından talepte bulunan kullanıcının sağladığı bilgileri ile geçerli bir token bilgisini olup olmadığını kontrol edilir.

```
checkAccessControl: function () {
    var hashedToken = App.getToken(userKey, serviceNo, SP);
    var r = `0x${hashedToken.slice(0, 64)}`
    var s = `0x${sihashedTokeng.slice(64, 128)}`
    var v = web3.toDecimal(hashedToken.slice(128, 130)) + 27
    App.contracts.Authenticator.deployed().then(function (instance) {
      return instance.authenticateUser.call(hashedToken, v, r,
s).then(function (result) {
        if (result == userKey) {
          alert("Erişim token'i doğrudur. Kullanıcı istenilen bilgiye
erişebilir !")
        }
      }).catch(function (err) {
        console.error(err);
      });
    });
  }
```

**Şekil 3-8:** Erişim token bilgisini kontrol fonksiyonu.

(3) Akıllı sözleşme tarafından yapılan token kontrol işleminin sonucu servis sağlayıcı düğümünü bildirilir.

(4) Akıllı sözleşmeden olumlu bir sonuç alınırsa, kullanıcının ilgili servisin bilgisine erişmesine sağlanacaktır.

# 4. BULGULAR

## 4.1. SİSTEM BİLEŞENLERİNİN GELİŞTİRİLMESİ

### 4.1.1. Ethereum Blockchain Ağının Geliştirilmesi

Konsorsiyum Ethereum Blockchain ağını oluşturmak için *Geth* aracı kullanılmıştır. *Geth*, tam bir Ethereum düğümü oluşturmamızı ve çalıştırmamızı sağlayan bir araçtır. Bununla beraber, geliştirme amaçlı yerel Blockchain ağını yaratmamızı sağlamaktadır.

Blockchain ağı oluşturmak için ilk önce, bu ağın başlangıç davranışını tanımlayacak olan *Genesis Block* oluşturulması gerekmektedir. Şekil 4.1'de *Genesis Block* yapısı gösterilmektedir.

```
{
  "config": {
    "chainId": 4224,
    "homesteadBlock": 0,
    "eip150Block": 0,
    "eip150Hash": "0x0000000000000000000000000000000000000000000000000000000000000000
0",
    "eip155Block": 0,
    "eip158Block": 0,
    "byzantiumBlock": 0,
    "constantinopleBlock": 0,
    "petersburgBlock": 0,
    "ethash": {}
  },
  "nonce": "0x0",
  "timestamp": "0x5db5da25",
  "extraData": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "gasLimit": "0x47b760",
  "difficulty": "0x80000",
  "mixHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "coinbase": "0x0000000000000000000000000000000000000000",
  "alloc": {},
  "number": "0x0",
  "gasUsed": "0x0",
  "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000"
}
```

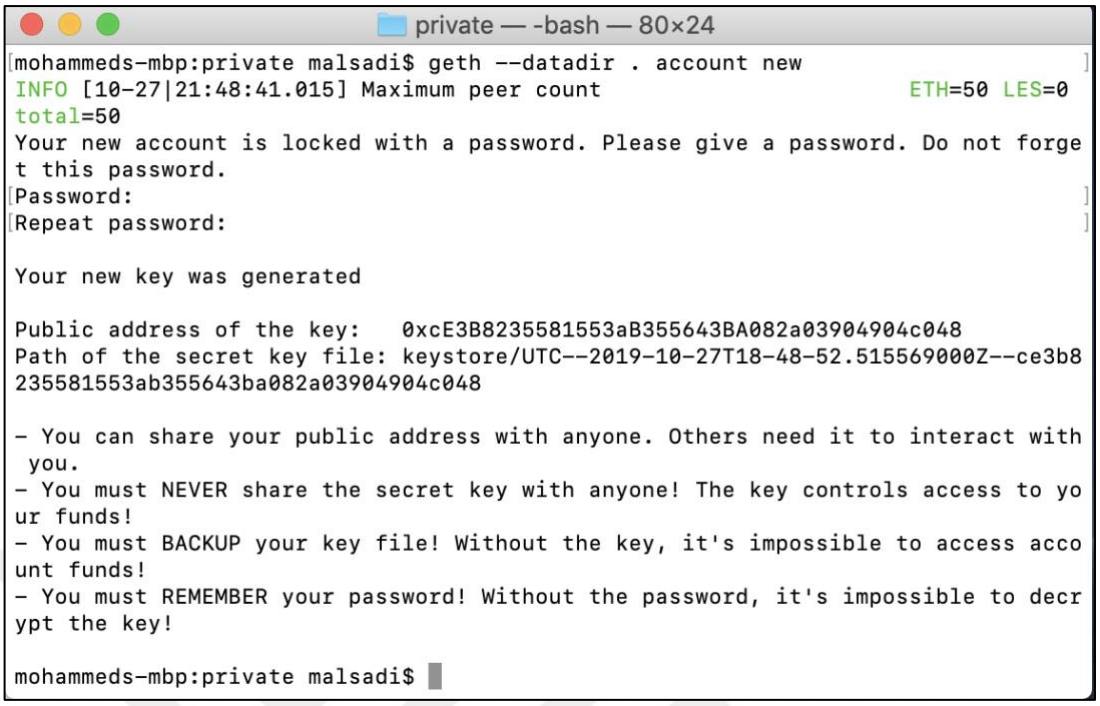
**Şekil 4-1:** Geth aracı kullanarak oluşturulan Genesis blok.

Genesis bloğunun en önemli alanları Tablo 4.1'de açıklanmıştır. *Genesis Block* oluşturulduktan sonra, hesap oluşturulur. Her bir hesap için iki temel bilgi bulunmaktadır: (1) Hesap adresi, ağda bulunan diğer düğümler için kullanılmaktadır; (2) Özel anahtar bilgisi, hesabı yönetmek için gerekli olan anahtardır. Özel anahtarın kaybedilmesi durumunda, hesaba hiçbir şekilde ulaşılamayacaktır; bu sebeple özel anahtarların güvenli bir şekilde saklanması gerekmektedir.

Şekil 4.2'de *Geth* aracı kullanarak oluşturulan yeni özel anahtar bilgisi detayları gösterilmektedir.

**Tablo 4.1:** Genesis bloğun detayları.

| **Unsurlar** | **Açıklama** |
|---|---|
| chainId | Zincir ağı tanımlayıcısıdır. Ethereum *mainnet* için 1 kullanılır. Geliştirme amaçlı kullanılan Blockchain ağı için farklı numaralar kullanılabilir. |
| timestamp | Blok oluşturulma zamanıdır. Aynı zamanda zorluk derecesi güncellemek için EVM tarafında kullanılır. İki ardışık blok arasında, bu değer küçük olursa zorluk derecesi yüksek olacaktır. |
| gasLimit | Bir bloğun içerdiği kayıtların harcayabileceği en büyük *gas* ( ether para pirimin miktarı) değeridir, |
| difficulty | Blockchain bloğunun valide edilmesinin ne kadar zor olduğunu belirleyen unsurdur. Bulmacayı çözmek için bir doğrulayıcı düğümünün hash fonksiyonunu çalıştırmak için, zorunda kalacağı ortalama süre ile doğrudan ilişkilidir. |
| number | Bloğun numarasıdır. |
| gasUsed | Bloğun içinde toplam kullanılan *gas* miktarıdır. |
| alloc | Belirli cüzdan (wallet) adreslerine belli *ether* miktarı tahsis etmek için kullanılan bir bölümdür. |
| parentHash | Önceki bloğun hash değeridir. |




**Şekil 4-2:** Geth aracı kullanarak Ethereum hesap oluşturma süreci.

### 4.1.2. Akıllı Sözleşmenin Geliştirilmesi

Akıllı Sözleşmeler, önerilen modelin çok önemli bir parçasıdır. Düğümler arasındaki kimlik doğrulama ve erişim kontrolü mekanizması, Akıllı Sözleşme tarafından kontrol edilmektedir. Geliştirilen Akıllı Sözleşme program kodu Şekil 4-3'te gösterilmektedir. Akıllı Sözleşme programı geliştirilecek olan Blockchain tabanlı sistemde, son kullanıcıların profillerinin oluşturulması, kullandıkları servislerin takip edilmesi, kullanıcıların kullanılmış veya kullandıkları servisler ile ilgili bilgileri sorgulaması ve benzeri fonksiyonları sağlamaktadır.

```
pragma solidity ^0.4.18;

contract Profiles {

struct AuthTokens
     {
        bytes32 hashedToken;
        address issuer;
        address user;
        string serviceNo;
     }

mapping(bytes32 => AuthTokens) tokenStructs;
bytes32[] tokenList;

struct Service
   {
     string serviceNo;
     address serviceOwner;
     string serviceDescription;
     address[] serviceUsers;
     uint256 servicePrice;
     address[] accessList;
   }

mapping(string => Service) serviceStructs;
string[] serviceList;

struct Users
   {
     address userID;
     address creatorID;
     string creationTime;
   }
mapping(address => Users) UsersStructs;
address[] usersList;

// events
event newUserEvent(bool result, address creator, string cTime, address userID);
event newServiceEvent(bool result, address creator, string servisNo);
event newAccessTokenEvent(bool result, address creator, address user, bytes32 token, string
serviceNo);
event useServiceEvent(bool result, address user, string servisNo);

function newServis(string _sno, string _desc, uint256 _price) public
    returns (bool success)
      {
        serviceStructs[_sno].serviceNo = _sno;
        serviceStructs[_sno].serviceDescription = _desc;
        serviceStructs[_sno].servicePrice = _price;
        serviceStructs[_sno].serviceOwner = msg.sender;
        serviceList.push(_sno);
        newServiceEvent(true, msg.sender, _sno);
        return true;
      }

function getService(string _sno) public view
    returns(string _serviceNo, address _owner, string _description, uint256 _price, uint
serviceUsers, uint accessCount)
      {
        return(serviceStructs[_sno].serviceNo, serviceStructs[_sno].serviceOwner,
serviceStructs[_sno].serviceDescription, serviceStructs[_sno].servicePrice,
        serviceStructs[_sno].serviceUsers.length, serviceStructs[_sno].accessList.length);
      }

function getServiceCount()
    public view
    returns(uint serviceCount)
      {
          return serviceList.length;
      }

function getServiceAtIndex(uint row)
    public view
    returns(string serviceIdentifier)
      {
         return serviceList[row];
      }
```

```
function getServiceUsers(string _sno) public view
returns(address[] usersList){
    return(serviceStructs[_sno].serviceUsers);
}

function getServiceAccessList(string _sno) public view
returns(address[] accessList){
    return(serviceStructs[_sno].accessList);
}

function useService(string _sno) public payable returns(bool result){

  require(msg.sender != serviceStructs[_sno].serviceOwner);
  require(msg.value == serviceStructs[_sno].servicePrice);
  serviceStructs[_sno].serviceOwner.transfer(msg.value);
  serviceStructs[_sno].serviceUsers.push(msg.sender);
  useServiceEvent(true, msg.sender, _sno);
}

function addAccessToService(string _sno, address _user, bytes32 uToken) public
    returns(bool success)
      {
        tokenStructs[uToken].hashedToken = uToken;
        tokenStructs[uToken].issuer = msg.sender;
        tokenStructs[uToken].user = _user;
        tokenStructs[uToken].serviceNo = _sno;
        tokenList.push(uToken);
        newAccessTokenEvent(true, msg.sender, _user, uToken, _sno);
        return true;
      }

 function getAccessToken(address _uID, address serviceProvider, string _sno, bytes32 _token)
    public view  returns(bytes32 uToken)
      {
        return(tokenStructs[_token].hashedToken);
      }

function getAccessTokenObject(bytes32 _token) public view
    returns(address user,address owner, string sno, bytes32 userToken){
      return(tokenStructs[_token].user, tokenStructs[_token].issuer,
tokenStructs[_token].serviceNo, tokenStructs[_token].hashedToken);
    }
function newUser(address _uID, string _ctime) public
    returns (bool success)
      {
        UsersStructs[_uID].userID = _uID;
        UsersStructs[_uID].creatorID = msg.sender;
        UsersStructs[_uID].creationTime = _ctime;
        usersList.push(_uID);
        newUserEvent(true, msg.sender, _ctime, _uID);
        return true;
      }

function getUser(address _uID) public view
    returns(address uID, address _creatorID, string _ctime)
      {
        return(UsersStructs[_uID].userID, UsersStructs[_uID].creatorID,
UsersStructs[_uID].creationTime);
      }
}
```

**Şekil 4-3:** Akıllı Sözleşme kodu.

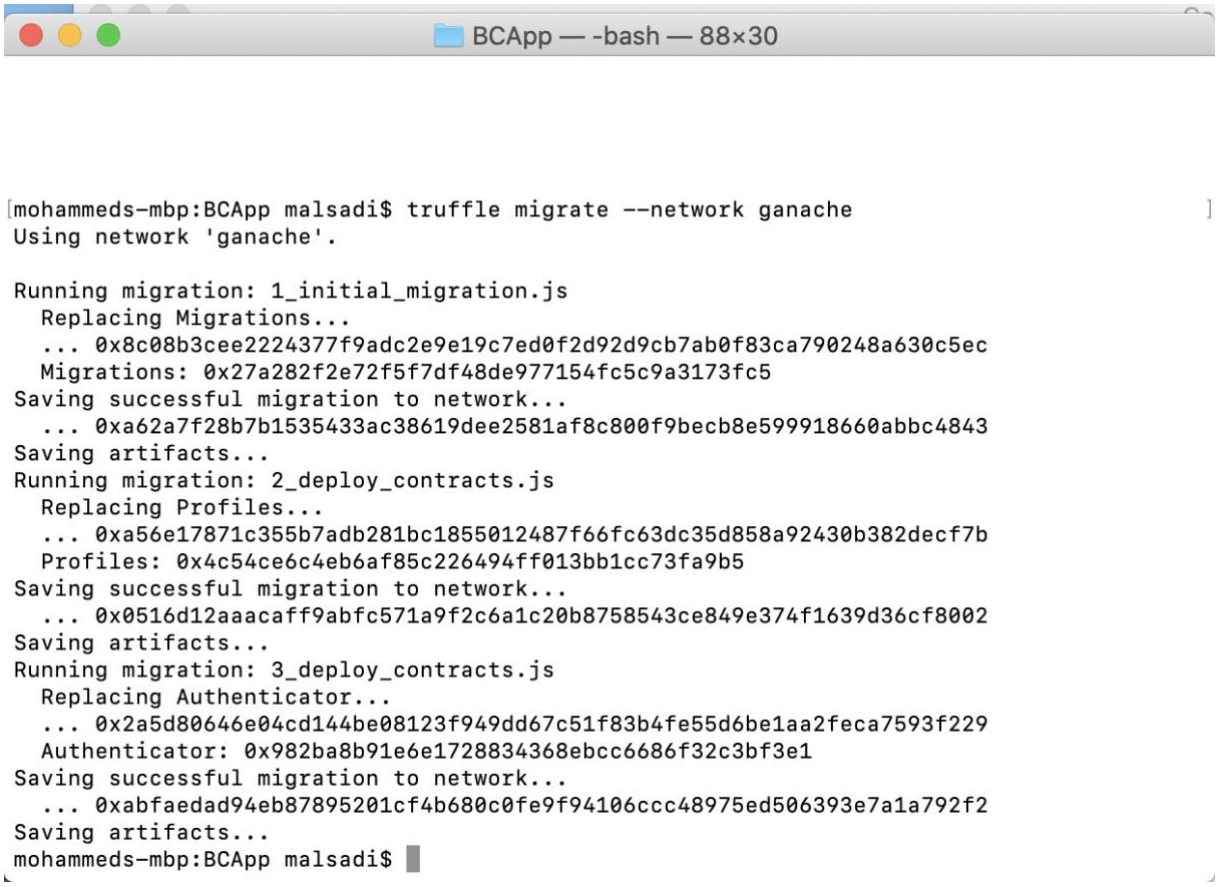
Şekil 4-3'te gösterildiği üzere *struct*, Akıllı Sözleşme için önemli bir veri tipidir. Nesne olarak saklayıp üzerinde kolayca işlem yapmak istenilen bir verinin *struct* olarak barındırılması gerekmektedir. Önerilen model kapsamında kullanılacak Akıllı Sözleşme için üç temel *struct* geliştirilmiştir:

- *User* struct, sisteme katılan son kullanıcıların bilgilerini tutmaktadır.
- *Service* struct, eklenen servislerin detaylarını içermektedir.
- *AuthToken* struct, belirli bir servisin bilgilerine erişim izni alan kullanıcıların adresi ve onlara verilen *token* bilgileri kaydedilmektedir.

Akıllı Sözleşme ile iletişim, Akıllı Sözleşme içerisinde bulunan ilgili fonksiyonun çağrılmasıyla gerçekleştirilmektedir. Önerilen model kapsamında geliştirilen Akıllı Sözleşmenin, önemli bazı fonksiyonları şu şekildedir:

- *newService* fonksiyonu: Servis sağlayıcısının yeni servis eklemesini sağlayacaktır.
- *newUser* fonksiyonu: Sisteme katılmak isteyen kullanıcının kimlik doğrulamasının olumlu sonuçlandırılmasından sonra, ilgili servis sağlayıcısının kullanıcı için profil oluşturmasını sağlayacaktır.
- *addAccessToService*: Sistemde bulunan bir servisin kullanım detaylarına veya bilgilerine erişmek için, bir kullanıcıya ilgili servise erişme hakkı tanımlamayı sağlayacaktır.

İlgili Akıllı Sözleşme kodunun, geliştirilen Blockchain ağına uygulanması için satır komutundan (Command Line, CMD) *Truffle* aracı çalıştırılmaktadır. Şekil 4.4'te gösterildiği gibi, *truffle migrate - -network ganache* komutu ile komut satırı üzerinde *Truffle* aracı kullanılmaktadır.




**Şekil 4-4:** Akıllı Sözleşme kodunun Truffle aracı ile Blockchain'e uygulanması.

### 4.1.3. Dağıtık Uygulamanın Geliştirilmesi

Sistem aktörlerinin Blockchain ağı ile iletişim kurmasını DApp sağlayacaktır. DApp üzerinden Blockchain ağına erişmek için ilk önce ağ ayarlarının tanımlanması gerekmektedir. Şekil 4.5'te gösterildiği gibi, her Blockchain ağı için ağ isminin, host adının, sahip olduğu port numarası ve benzeri detayların tanımlanması gerekmektedir.

```
module.exports = {
  // See <http://truffleframework.com/docs/advanced/configuration>
  // for more about customizing your Truffle configuration!
  networks: {
    ganache: {
      host: "127.0.0.1",
      port: 7545,
      network_id: "*",
      gas: 4700000 // Match any network id
    },
    develop: {
      port: 8545
    }
  }
};
```

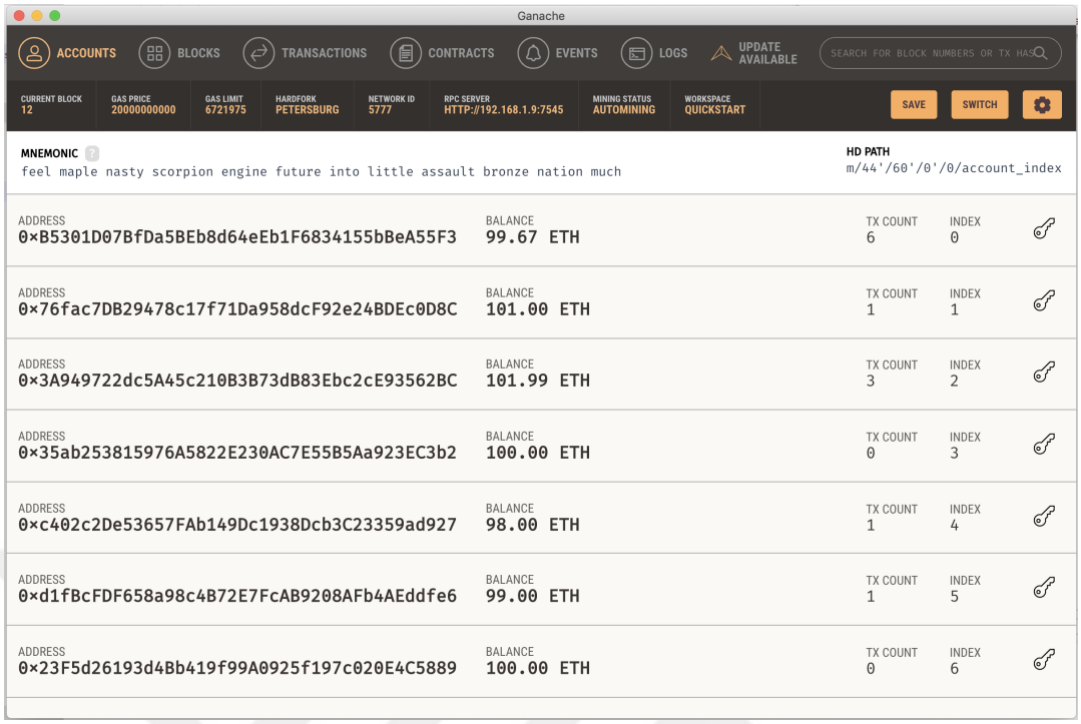
**Şekil 4-5:** Blockchain ağına erişmek için DApp üzerinde ağ ayarlanması.

Gerekli ayarlar ve tanımlamalar yapıldıktan sonra, Blockchain ağına yüklenen Akıllı Sözleşmenin fonksiyonlarına erişmek ve kullanmak amacıyla, önyüz tarafı olarak web uygulaması - web sayfaları - geliştirilmiştir. Web sayfaların lokal olarak çalıştırılması için *lite-server* sunucu uygulaması kullanılmıştır.

## 4.2. SİSTEM KULLANIM SENARYOSU VE UYGULAMA

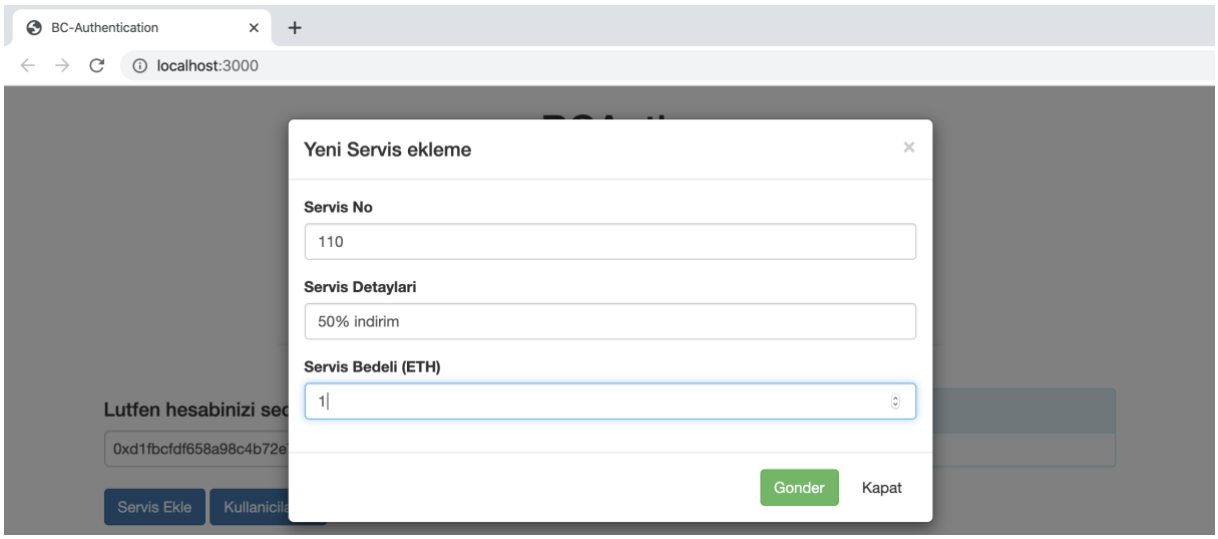
Çalışma kapsamında geliştirilen Blockchain ağı ve Akıllı Sözleşme programının gerçekleştirilmesi amacıyla, bir bilgi sorgulama senaryosu uygulanmıştır. Bu uygulama, Blockchain ağında bulunan düğümlerin, sunulan servisler ile ilgili bilgileri sorgulama ve erişme işlemleri ile ilgilidir. Son kullanıcının servisi kullanması, sadece servis sağlayıcı tarafından belirlenen bilgilere erişmesi anlamına gelmektedir.

Bilgi sorgulama senaryosu üzerinden açıklanan işlemleri ve Blockchain ile ilgili diğer işlemleri takip etmek için Şekil 4-6'da gösterildiği üzere *Ganache* aracı kullanılmıştır. Hem düğümlerin bilgisine hem de Blockchain blokları ve kayıtlarına *Ganache* arayüzü üzerinden kolayca ulaşılmaktadır.




**Şekil 4-6:** Ganache uygulamasının arayüzü.

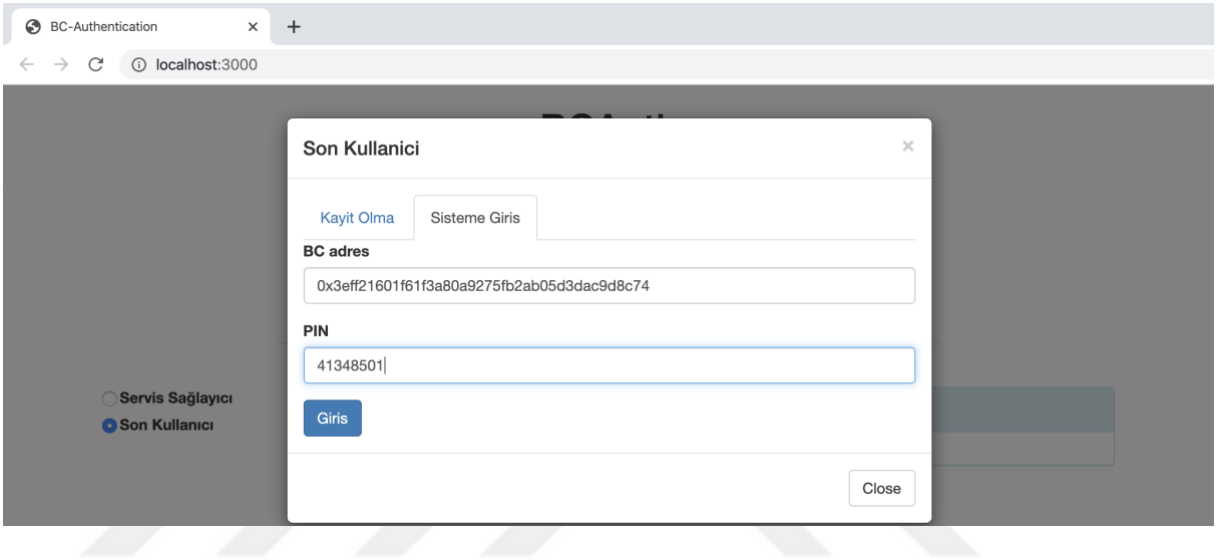
Öncelikle servis kullanımının uygulanması için, servis sağlayıcılarının Ethereum Blockchain ağında, sunmak istediği her servis için Şekil 4.7'de gösterildiği üzere bir profil oluşturması gerekmektedir. Blockchain ağı üzerinde saklanmak üzere servis numarası, detaylar, ücreti ve benzeri bilgileri içeren bir servis tanımlaması gerçekleştirilecektir.




**Şekil 4-7:** Yeni servis ekleme işleminin uygulama arayüzü.

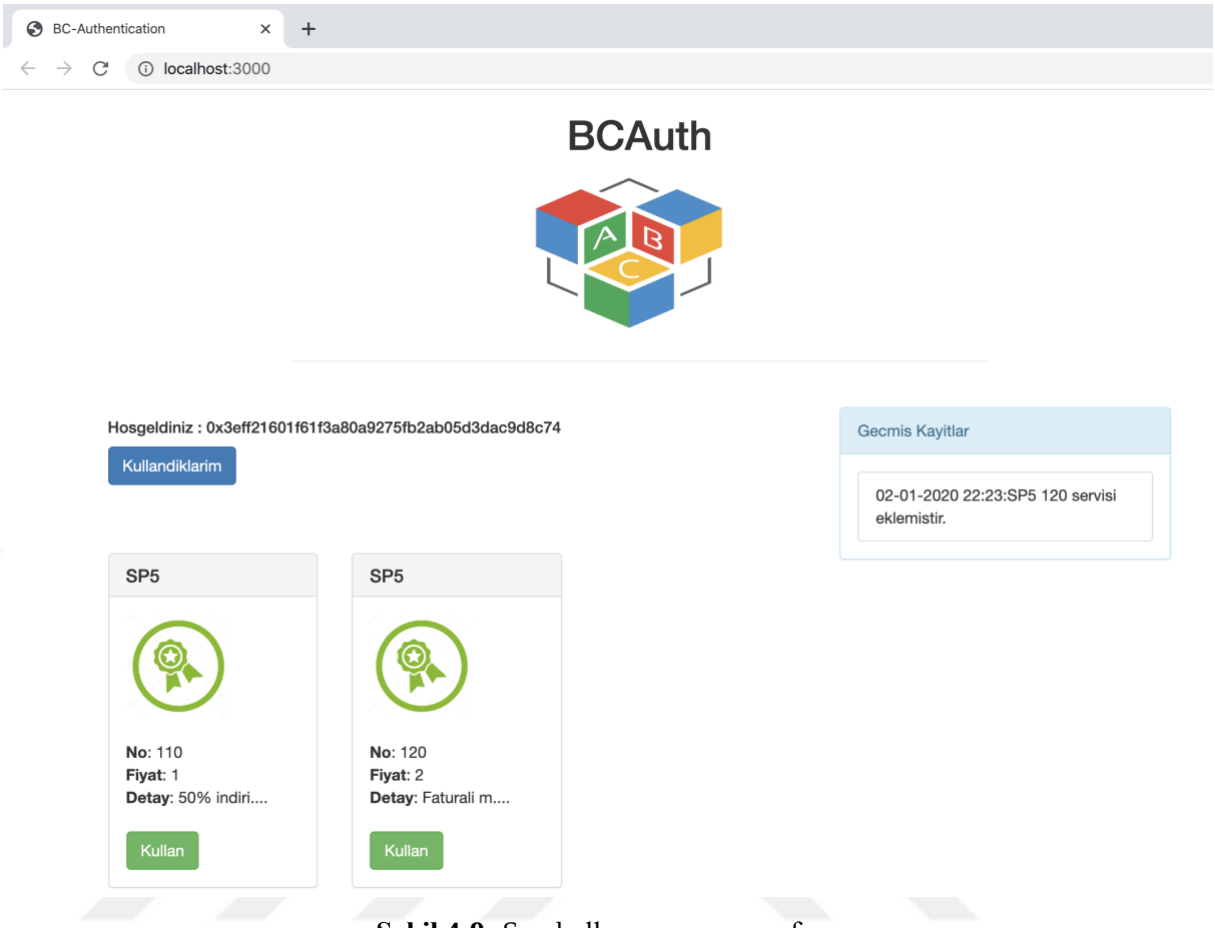
Şekil 4-14'de servis kullanımı akışının sistem sıra diyagramı gösterilmektedir. Sistem kullanımı uygulaması kısaca şu adımlardan oluşmaktadır:

(1) Sisteme kayıt aşamasını başarı ile tamamlayan kullanıcı -Bölüm 3.2.2'de anlatıldığı üzere- şekil 4-8'de gösterildiği üzere Blockchain adresi ve sahip olduğu PIN bilgisi ile sisteme giriş yapar.




**Şekil 4-8:** Son kullanıcının sisteme giriş arayüzü.

(2) Kullanıcının girdiği bilgileri doğru ise, servis sağlayıcı düğümler tarafından eklenen tüm servisleri kullanıcıya sunulacaktır.



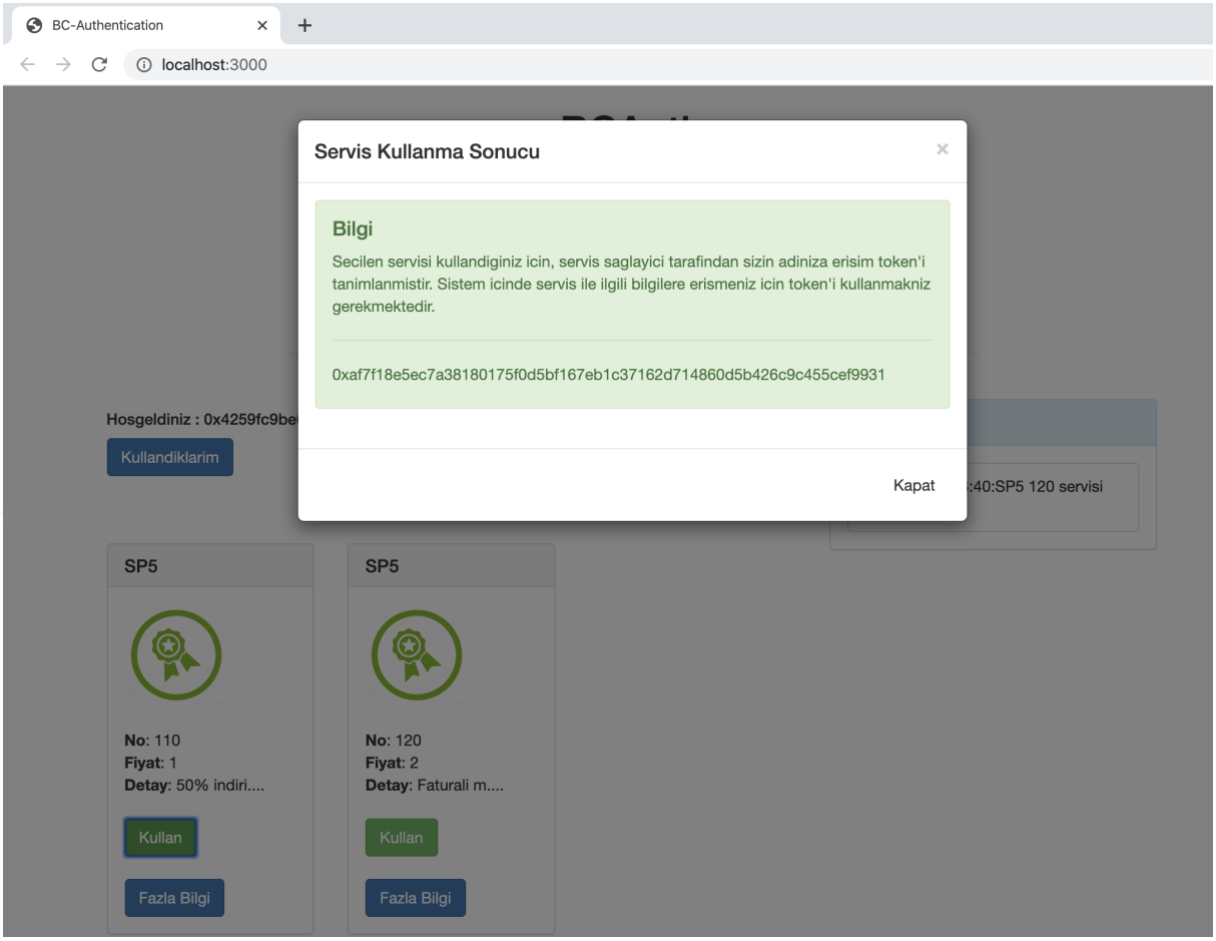

**Şekil 4-9:** Son kullanıcının ana sayfası.

(3) Son kullanıcı, mevcut servislerden kullanmak istediği servisi seçip *kullan* butonu tıklayarak kullanma talebi Blockchain ağına gönderir.

(4) Kullanma talebi onaylanması için Akıllı Sözleşmeye aktarılır. Kullanma talebi, Servis sağlayıcı tarafından belirlenen kullanma şartlara uygun olduğu takdirde onaylanır.

(5) Servis sağlayıcı, kullanma talepte bulunan son kullanıcıyı servisi kullanan listesine ekler ve son kullanıcının servis ile ilgili diğer bilgilere erişmesi için erişim token değeri oluşturur.

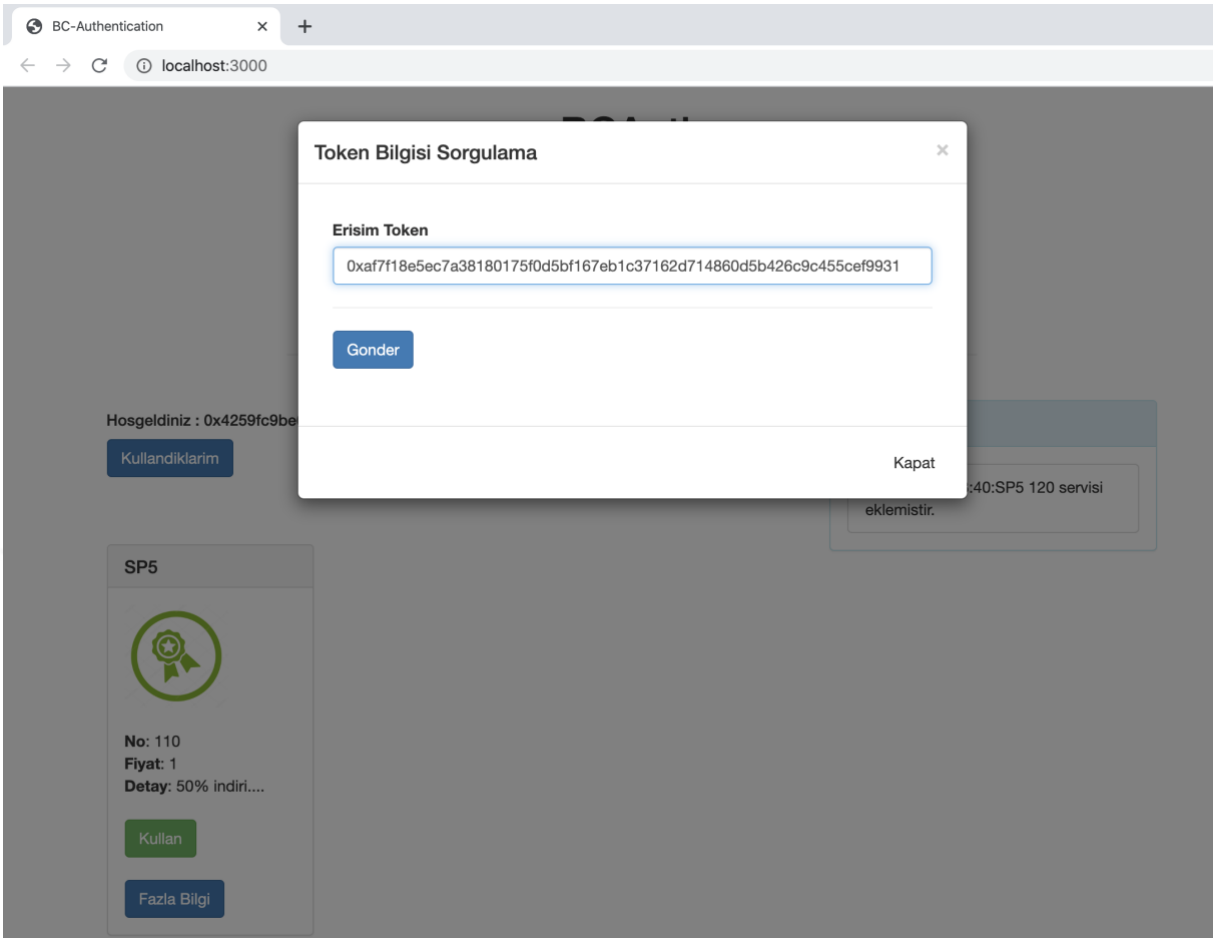
(6) Servisi kullanma talebin sonucu şekil 4-10'da gösterildiği üzere kullanıcıya iletilir.




**Şekil 4-10:** Servisi kullanma talebinin sonucunu kullanıcıya bildirimi arayüzü.

Son kullanıcılar, servisi kullanma dışında servis ile ilgili verilere erişmek ister. Servis ile ilgili verilere erişmek isteyen bir kullanıcının erişim *token* sağlaması gerekmektedir. Servis bilgilere erişim uygulaması kısaca şu adımlardan oluşmaktadır:

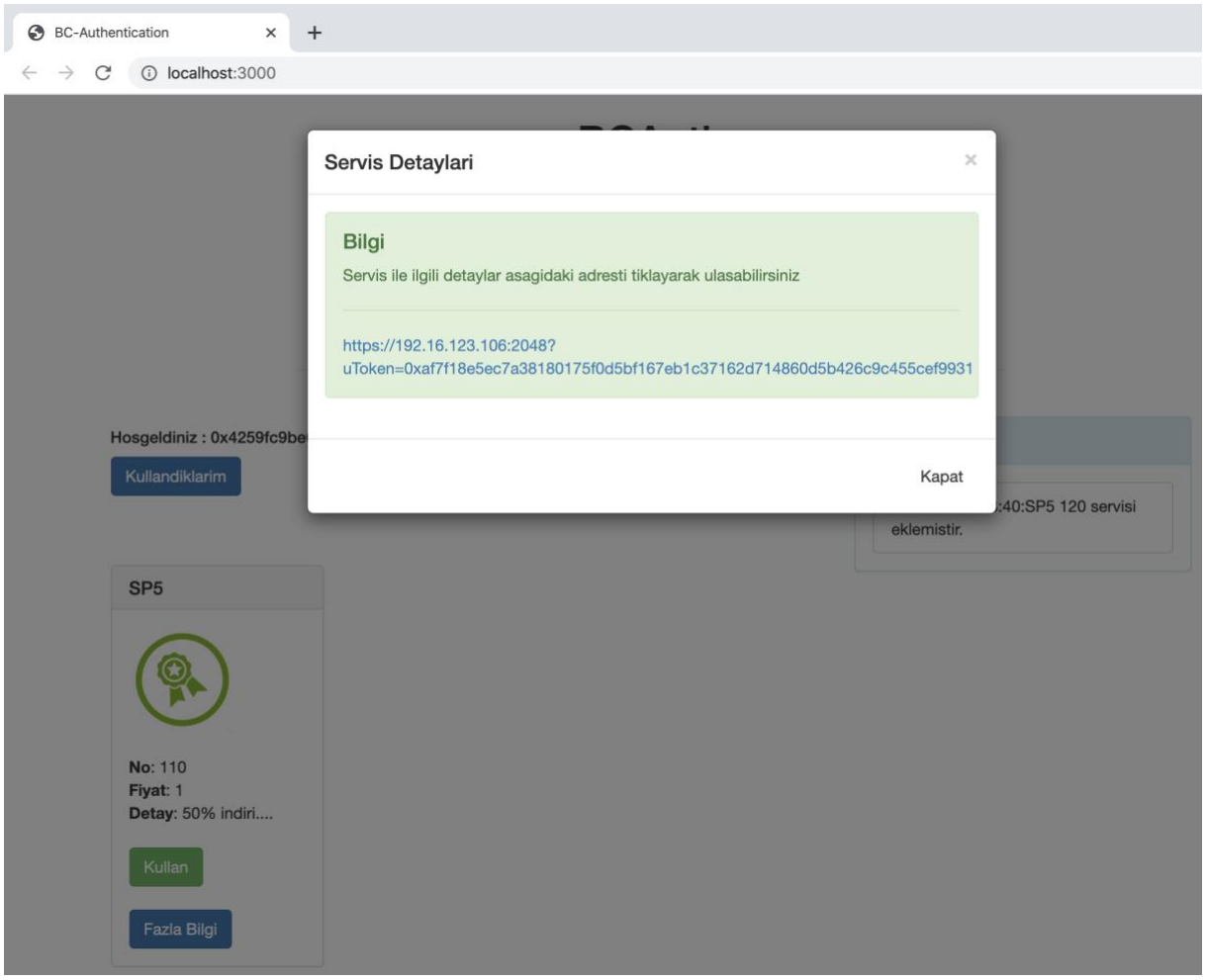
(1) Servis ile ilgili verilere erişim talepte bulunan kullanıcının kendi adresini ve sahip olduğu token değeri kullanarak erişim hakkı olup olmadığını kontrol etmek amacıyla, Akıllı Sözleşme tarafından sorgulaması yapılır.



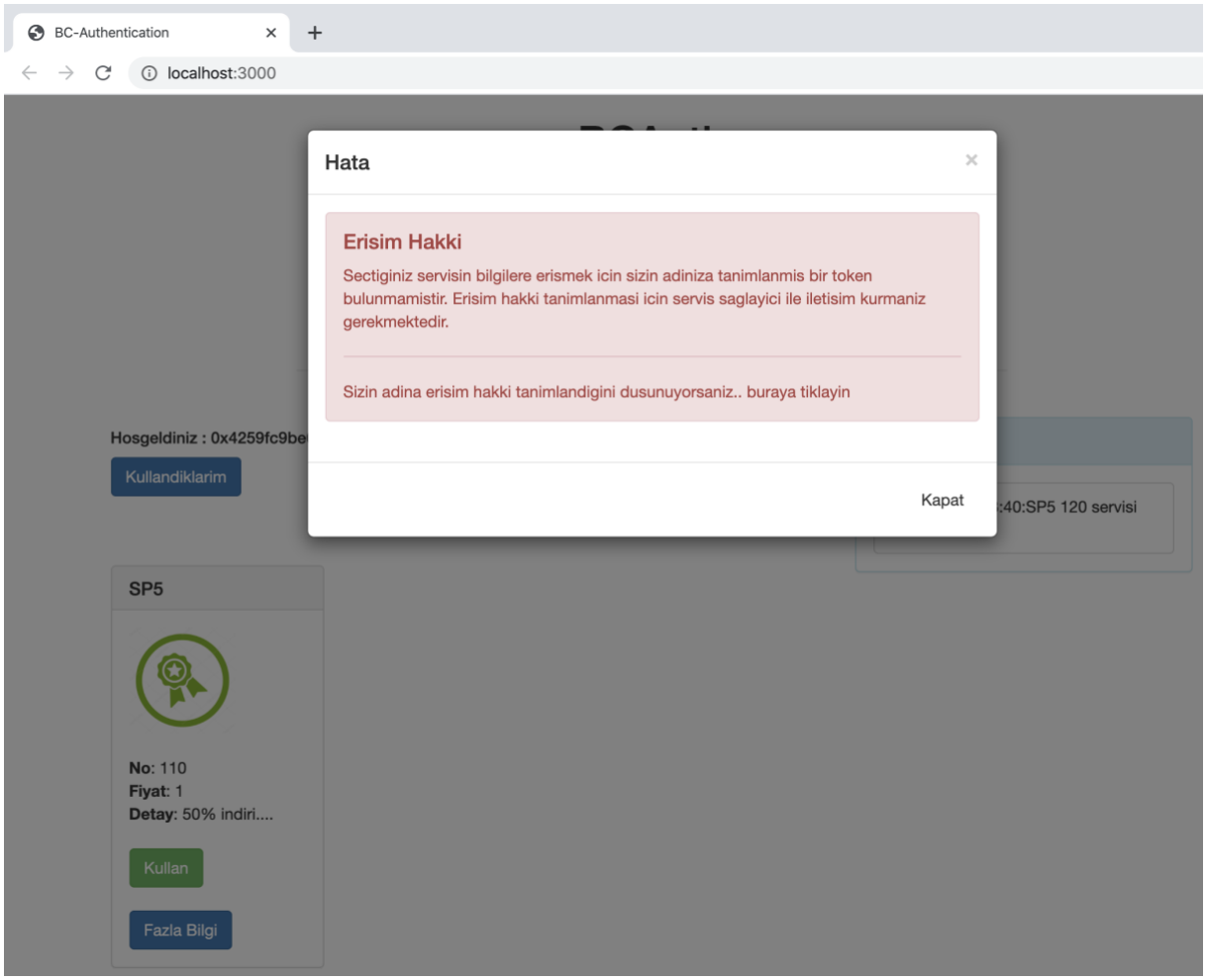

**Şekil 4-11:** Servis verilerine erişmek için kullanıcının kullandığı arayüzü.

(2) Akıllı Sözleşme, son kullanıcı tarafından sağlanan bilgileri ile servis numarası ve servis sağlayıcı adres bilgisi kullanarak *token* sorgulama yapar.

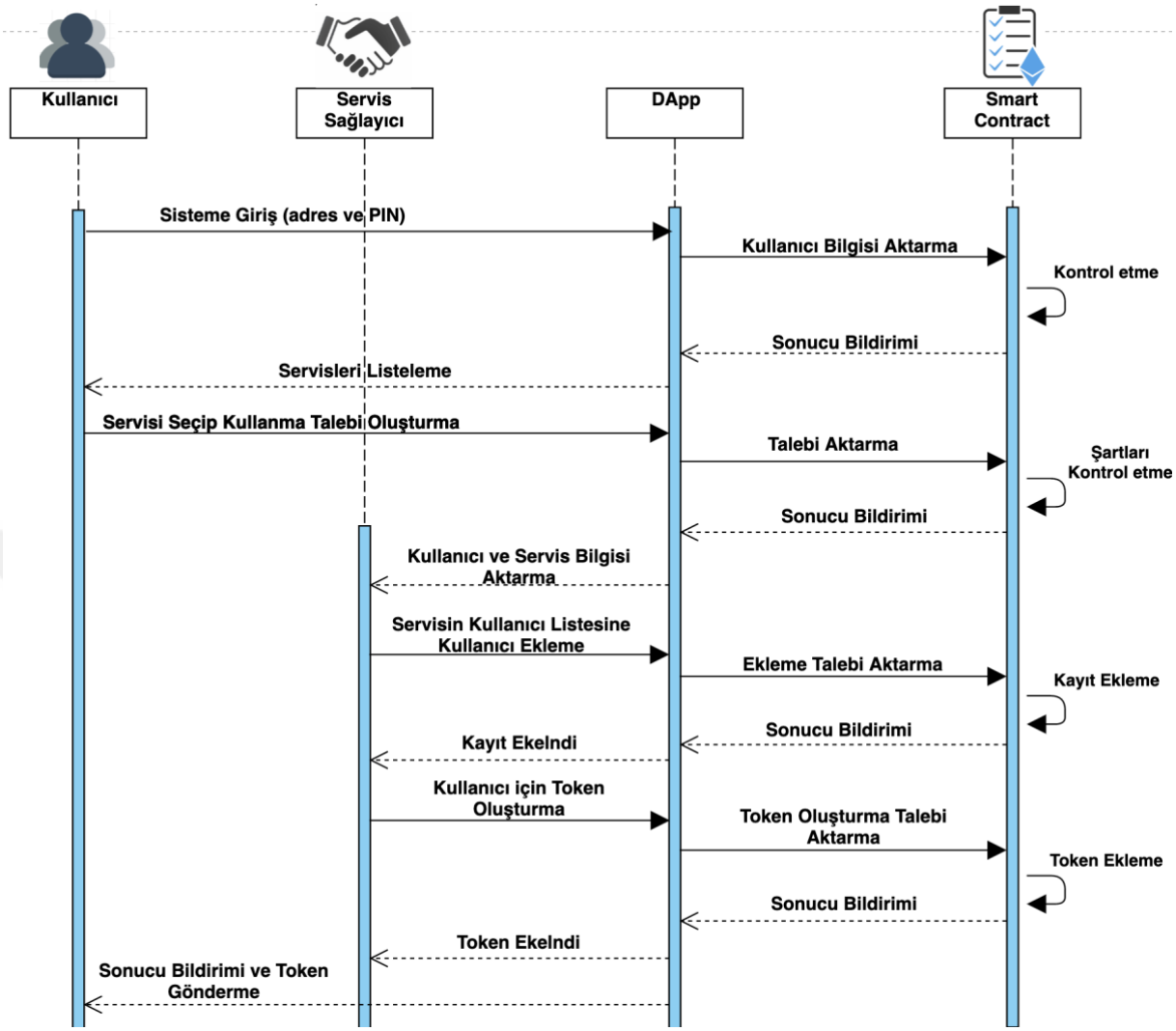
(3) *Token* değerinde kayıtlı olan kullanıcının Blockchain adresi ve erişim izni geçerlilik süresi alanları kontrol edilir. Gönderici düğümün adresi ile *token* değerde bulunan Blockchain adres bilgisi ile aynı ve *token* değerinin geçerlilik süresi dolmamış ise, -Şekil 4-12'de gösterildiği üzere- kullanıcıya talep ettiği bilgi verilir. *Token* değeri geçerli değilse veya adres bilgisi karşılaştırması olumsuz ise -Şekil 4-13'te gösterildiği üzere- kullanıcının bilgiye erişim talebi isteği reddedilir.




**Şekil 4-12:** Kullanıcının erişim talebini onaylama arayüzü.




**Şekil 4-13:** Kullanıcının erişim talebini reddetme arayüzü.




**Şekil 4-14:** Servis kullanım sistem sıra diyagramı.

## 4.3. DEĞERLENDİRME

Kimlik doğrulama, bir sistemin güvenliği sağlanması için en temel gereksinimlerden biridir. Bu çalışmada, kullanıcının kayıt olma aşamasında kimliğinin doğrulanması ve ilgili servis sağlayıcı tarafından kullanıcıya ait güvenilir bir profil oluşturulması önerilmiştir. Kayıt olma aşamasında, kullanıcıdan istenilen bilgiler ile ilgili servis sağlayıcı düğümü tarafından kullanıcıya ait bir kayıt akıllı sözleşmeye eklenecektir. Eklenen kayıt, kullanıcının açık olan bilgilerini (Blockchain adresi, profil oluşturulma zamanı ve profili oluşturan servis sağlayıcı) içermektedir. Bu kayıt daha sonra diğer düğümler tarafından da kullanılabilmektedir. Bu bilgiler haricindeki, kullanıcının profili ile ilgili hassas ve gizli olan bilgiler telefon üzerinden kullanıcıya iletilmesi sağlanacaktır.

Önerilen model ile, Konsorsiyum Blockchain sistemine katılacak olan son kullanıcıların kimlik doğrulama işlemi iki aşamalı bir süreç tasarlanmıştır. İlk aşamadaki iletişim son kullanıcı ile servis sağlayıcı arasında gerçekleştirmektedir. Bugün evrensel olarak kabul edilen, üç kimlik doğrulama faktörü bulunmaktadır: bildikleriniz (şifre), sahip olduklarınız (ATM kartı veya mobil cihazı) ve ne olduğunuz (parmak izi) [109]. Örneğin İki Faktörlü Kimlik Doğrulama [110] (Two Factor Authentication) yönteminde, kullanıcıların kimlik doğrulaması iki doğrulama faktörü (genelde bildiğiniz ve sahip olduğunuz) kullanılarak yapılmaktadır. Bu çalışma kapsamında önerilen Blockchain tabanlı kimlik doğrulama modelinde iki faktör kimlik doğrulama yöntemi kullanılmaktadır. Birinci aşamada, OTP uygulaması ile kullanıcının sahip olduğu mobil cihaz üzerinden kimlik doğrulaması tasarlanmıştır, ve ardından ikinci aşamada, son kullanıcıların gerçek kimliği, üçüncü bir güvenilir taraf olarak devlet kuruluşu üzerinden kullanıcı kimlik bilgilerinin çapraz sorgulama ile doğrulaması önerilmiştir.

Çalışma kapsamında önerilen model, aynı zamanda erişim kontrol mekanizmaları açısından da bir katkı sağlamaktadır. Birbirine güvenmeyen aktörlerin, Blockchain üzerinde güvenli bir ekosistemde altında veri paylaşımını sağlamaktadır. Çalışma kapsamında yapılan uygulama ile, akıllı sözleşme kullanacak modelin, dinamik bir yapıya sahip olması sağlanmıştır. Uygulamada görüldüğü üzere, servis sağlayıcıların sundukları servisler, Akıllı Sözleşme üzerinden

yapılmaktadır. Bu bağlamda, kullanıcı tarafından gönderilen bilgi sorgulama ve benzeri talepler Akıllı Sözleşme tarafından kontrol edilmektedir. Akıllı Sözleşme üzerinde tanımlanan erişim kurallarına göre, servis sağlayıcılar oluşturdukları servisler ile ilgili bilgilerin erişimini sağlamaktadır. Tanımlanan kurallar, servis sağlayıcı tarafından oluşturulup *token* bilgisi ile Akıllı Sözleşmeye kaydedilmektedir. Hem kullanıcı hem de servis ile ilgili bazı bilgileri oluşturulan *token* bilgisinin özeti (hash) olarak kaydetmektedir.

Tez çalışmasında önerilen Blockchain tabanlı kimlik doğrulama modelinin çeşitli alanlarda ve iş modelleri kapsamında uygulanması mümkündür. Modelin bir bütünlük içerisinde sağlayacağı kimlik doğrulama ve erişim kontrol mekanizması özellikleri sayesinde, servis sağlayıcılar arasında güvenilir bir iletişim ortamının oluşması ve iş birliği gelişimi öngörülmektedir.

Günümüzde yaygın olarak karşılaştığımız bir iş modeli olarak, belirli bir servis sağlayıcısının müşterisi olan bir kullanıcının, anlaşmalı başka bir servis sağlayıcısından hizmet alması karşılığında çeşitli kampanyalar, indirimler veya faydalar kazanması sağlanmaktadır. Örneğin, bir havayolu firmasına ait bir müşteri, havayolu firmasının anlaşmalı olduğu otellerde konaklaması halinde daha uygun fiyatlar veya düşük maliyetler ile hizmet alabilir. Bu tarz avantajlarından faydalanmak isteyen kullanıcıların, ilgili servis sağlayıcısının müşterisi olduğunu ispatlaması gerekmektedir. Üyelik kartı veya e-posta çıktısını gösterme en yaygın kullanılan yöntemlerdir; fakat bu tarz yöntemler kullanıldığı zaman servis sağlayıcısının kullanıcıların sunduğu ispatı teyit etmesi gerekmektedir. Bir ekosistemde yer alan servis sağlayıcıların -her birinin kendine ait özel bir bilgi sistemi kullandığı için- aralarında paylaşılan bilgilerin doğruluğunun kontrol edilmesi, yoğun iletişim ve zaman gerektirir. Bu durum zorlayıcı bir süreç olabilir.

Tez kapsamında önerilen model ile, Blockchain ağı üzerinde yapılan herhangi bir işlem tüm düğümler -servis sağlayıcılar- tarafından anında görüntülenebilecektir. Dolayısıyla, ekstra adımlara gerek kalmadan güvenli veri iletişimi ve veri bütünlüğü sağlanabilecektir. Akıllı Sözleşmelerin de kullanımıyla beraber sistemlerin daha dinamik bir yapıya sahip olması mümkün olacaktır.

# 5. TARTIŞMA VE SONUÇ

Tez çalışmasının amacı, akıllı ortamlarda Blockchain teknolojisinin özelliklerinden faydalanarak güvenli bir kimlik doğrulama modelin mimarisi sağlamaktadır. Bu mimarisi daha dinamik bir yapıda olabilmesi için Smart Contract altyapısı ile sağlanmaktadır. Bu kapsamda tez çalışmasında akıllı ortamlarda yer alan ve birbirine güvenmeyen aktörler arasında güvenli bir iletişim altyapısı oluşturulmuştur.

Blockchain tabanlı kimlik doğrulama modeli, akıllı ortamlarda yer alan aktörler arasında merkezi olmayan yapı üzerinde güvenli bir iletişim modeli sağlamaktadır. Aktörler arasında paylaşılan bilgileri Blockchain üzerinde saklayabilmesi, bu bilgileri kullanabilmesi ve yönetebilmesi sağlanmıştır. Aynı zamanda aktörlerin gerçekleştirdiği işlemleri bir web uygulaması üzerinden takip edebilmektedir. Sistem dışındaki bireysel kullanıcıların, iki aşamalı kimlik doğrulama mekanizması sayesinde sisteme dahil olabilmeleri sağlanmıştır.

Belirli bir servis sağlayıcı üzerinden kimliğini doğrulayan bir kullanıcı sistemdeki diğer servis sağlayıcılardan ekstra herhangi bir işleme gerekmeden servisi talep edebilir. Bu nedenle, Konsorsiyum Blockchain ağında yer alan diğer servis sağlayıcılardan servis talep edildiğinde kimlik doğrulama işlemini bir daha yapılmayacaktır. Bu özelliği sayesinde belirli servis sağlayıcının üyesi olan bir kullanıcı ağdaki diğer servis sağlayıcıların sunabileceği indirim veya kampanyalardan kolayca faydalanabilmektedir.

Blockchain tabanlı kimlik doğrulama modelinin diğer önemli bir katkısı, Smart Contract ile erişim mekanizması oluşturmaktadır. Akıllı ortamlarda yer alan aktörlerin hassas olan tanımladıkları bilgilere, dinamik bir şekilde erişim kuralları tanımlayarak diğer aktörlerin ulaşılmasını sağlamaktadır.

Önerilen modelin farklı alanlarda uygulanması mümkündür. Birbirine güvenmeyen aktörler için uygun bir altyapı oluşturmayı hedefleyen bu model, Blockchain üzerinde aktörler arasında güvenli ve tutarlı bilgi alışverişi sağlamaktadır. Modelin bir bütünlük içerisinde sağlayacağı kimlik doğrulama ve erişim kontrol mekanizması özellikleri sayesinde, servis sağlayıcılar arasında güvenilir bir iletişim ortamının oluşması ve iş birliği gelişimi öngörülmektedir. Blockchain teknolojisinin sağladığı güvenlik seviyesi ile, aktörlerin ayrı bir güvenlik sistemi kullanmasına gerek kalmamaktır.

Önerilen modelde, çok yüksek seviyede hassas veriler paylaşılmaması durumunda iki faktör kimlik doğrulama (2FA) yöntemi yeterli olduğunu öngörülmektedir. Ödeme bilgileri gibi hassas verilerin paylaşılması durumunda ise, kimlik doğrulama yöntemine biyometrik faktörü eklenerek üç faktör doğrulama (3FA) yöntemi kullanılabilecektir.

Önerilen modelde, diğer önemli bir konu da, gerçek kimliği doğrulayan devlet kurumunun bir varlık olarak sisteme olmaması durumunda sistemdeki kimlik doğrulama yönteminin güvenliğinin arttırılması gerekmektedir. Sadece OTP (One-time-password) bilgisi ile son kullanıcının kimliğini doğrulamak yeterli değildir; dolayısıyla OTP ile birlikte başka bilgi kullanılması önem arz etmektedir. Yüz tanıma veya parmak izi gibi biyometrik verileri kullanılması sistem güvenliği açısından fayda sağlayacaktır. Biyometrik verilerin Blockchain sistemlerinde kullanılması diğer önemli bir konuyu gündeme getirmektedir: biyometrik verinin bu sistem kapsamında nasıl alınacağı ve nerede, nasıl bir biçimde saklanacağı konusudur.

# KAYNAKLAR


[1]. Nixon, P., Lacey, G. and Dobson, S. eds., 2012, Managing Interactions in Smart Environments: *1st International Workshop on Managing Interactions in Smart Environments (MANSE'99)*, Dublin, December 1999. Springer Science & Business Media.

[2]. Ryu, M., Kim, J. and Yun, J., 2015, Integrated semantics service platform for the Internet of Things: A case study of a smart office, *Sensors*, *15*(1), pp.2137-2160.

[3]. Kim, H.M. and Laskowski, M., 2018, Toward an ontology-driven blockchain design for supply-chain provenance, *Intelligent Systems in Accounting, Finance and Management*, *25*(1), pp.18-27.

[4]. Wollschlaeger, M., Sauter, T. and Jasperneite, J., 2017, The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0, *IEEE industrial electronics magazine*, *11*(1), pp.17-27.

[5]. Stefanov, D.H., Bien, Z. and Bang, W.C., 2004, The smart house for older persons and persons with physical disabilities: structure, technology arrangements, and perspectives, *IEEE transactions on neural systems and rehabilitation engineering*, *12*(2), pp.228-250.

[6]. Catarinucci, L., De Donno, D., Mainetti, L., Palano, L., Patrono, L., Stefanizzi, M.L. and Tarricone, L., 2015, An IoT-aware architecture for smart healthcare systems, *IEEE Internet of Things Journal*, *2*(6), pp.515-526.

[7]. Mohanty, S.P., Choppali, U. and Kougianos, E., 2016, Everything you wanted to know about smart cities: The internet of things is the backbone, *IEEE Consumer Electronics Magazine*, *5*(3), pp.60-70.

[8]. Li, S., Da Xu, L. and Zhao, S., 2015, The internet of things: a survey, *Information Systems Frontiers*, *17*(2), pp.243-259.

[9]. Whitmore, A., Agarwal, A. and Da Xu, L., 2015, The Internet of Things—A survey of topics and trends, *Information Systems Frontiers*, *17*(2), pp.261-274.

[10]. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. and Ayyash, M., 2015, Internet of things: A survey on enabling technologies, protocols, and applications, *IEEE communications surveys & tutorials*, *17*(4), pp.2347-2376.

[11]. Nixon, P., Wagealla, W., English, C. and Terzis, S., 2005, Security, privacy, and trust issues in smart environments.

[12]. Wang, J., Yang, Y. and Yurcik, W., 2005, July. Secure smart environments: Security requirements, challenges and experiences in pervasive computing, In *Experience Workshop on Pervasive Computing*.

[13]. Bertin, P., Bonjour, S. and Bonnin, J.M., 2009, Distributed or centralized mobility?, In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference* (pp. 1-6). IEEE.

[14]. De Filippi, P. and McCarthy, S., 2012, Cloud computing: Centralization and data sovereignty, *European Journal of Law and Technology*, *3*(2).

[15]. Atlam, H.F. and Wills, G.B., 2019, Intersections between IoT and distributed ledger, In *Advances in Computers* (Vol. 115, pp. 73-113). Elsevier.

[16]. Andreev, S., Galinina, O., Pyattaev, A., Gerasimenko, M., Tirronen, T., Torsner, J., Sachs, J., Dohler, M. and Koucheryavy, Y., 2015, Understanding the IoT connectivity landscape: a contemporary M2M radio technology roadmap, *IEEE Communications Magazine*, *53*(9), pp.32-40.

[17]. Whitmore, A., Agarwal, A. and Da Xu, L., 2015, The Internet of Things—A survey of topics and trends, *Information Systems Frontiers*, *17*(2), pp.261-274.

[18]. Barcelo, M., Correa, A., Llorca, J., Tulino, A.M., Vicario, J.L. and Morell, A., 2016, IoT-cloud service optimization in next generation smart environments, *IEEE Journal on Selected Areas in Communications*, *34*(12), pp.4077-4090.

[19]. Weiser, M., 1999, The computer for the 21st century, *ACM SIGMOBILE mobile computing and communications review*, *3*(3), pp.3-11.

[20]. Cook, D. and Das, S.K., 2004, *Smart environments: technology, protocols, and applications* (Vol. 43). John Wiley & Sons.

[21]. Nugent, C.D., McClean, S.I., Cleland, I. and Burns, W., 2014, Sensor Technology for a Safe and Smart Living Environment for the Aged and Infirm at Home.

[22]. Hilton, S., 2016, Dyn analysis summary of friday october 21 attack. *Dyn blog https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack*, [Ziyaret tarihi: 10 Aralık 2019]

[23]. Atzori, L., Iera, A. and Morabito, G., 2010, The internet of things: A survey, *Computer networks*, *54*(15), pp.2787-2805.

[24]. Liu, J., Xiao, Y. and Chen, C.P., 2012, Authentication and access control in the internet of things, In 2012 *32nd International Conference on Distributed Computing Systems Workshops* (pp. 588-592). IEEE.

[25]. Gupta, B.B. and Quamara, M., 2018, An identity based access control and mutual authentication framework for distributed cloud computing services in IoT environment using smart cards, *Procedia computer science*, *132*, pp.189-197.

[26]. Das, A.K., Wazid, M., Kumar, N., Vasilakos, A.V. and Rodrigues, J.J., 2018, Biometrics-based privacy-preserving user authentication scheme for cloud-based industrial Internet of Things deployment, *IEEE Internet of Things Journal*, *5*(6), pp.4900-4913.

[27]. Nakamoto, S. and Bitcoin, A., 2008, A peer-to-peer electronic cash system, *https://bitcoin.org/bitcoin.pdf*.

[28]. Crosby, M., Pattanayak, P., Verma, S. and Kalyanaraman, V., 2016, Blockchain technology: Beyond Bitcoin, *Applied Innovation*, *2*(6-10), p.71.

[29]. Swan, M., 2015, *Blockchain: Blueprint for a new economy*, " O'Reilly Media, Inc.".

[30]. Deshpande, A., Stewart, K., Lepetit, L. and Gunashekar, S., 2017, Distributed Ledger Technologies/Blockchain: Challenges, opportunities and the prospects for standards, *Overview report The British Standards Institution (BSI)*, pp.1-34.

[31]. Puthal, D., Malik, N., Mohanty, S.P., Kougianos, E. and Yang, C., 2018, The blockchain as a decentralized security framework [future directions], *IEEE Consumer Electronics Magazine*, *7*(2), pp.18-21.

[32]. Pilkington, M., 2016, Blockchain technology: principles and applications, In *Research handbook on digital transformations*. Edward Elgar Publishing.

[33]. Viriyasitavat, W. and Hoonsopon, D., 2019, Blockchain characteristics and consensus in modern business processes, *Journal of Industrial Information Integration*, *13*, pp.32-39.

[34]. Bozic, N., Pujolle, G. and Secci, S., 2016, A tutorial on blockchain and applications to secure network control-planes, In *2016 3rd Smart Cloud Networks & Systems (SCNS)* (pp. 1-8). IEEE.

[35]. Mylrea, M. and Gourisetti, S.N.G., 2017, Blockchain for smart grid resilience: Exchanging distributed energy at speed, scale and security, In *2017 Resilience Week (RWS)* (pp. 18-23). IEEE.

[36]. Angraal, S., Krumholz, H.M. and Schulz, W.L., 2017, Blockchain technology: applications in health care, *Circulation: Cardiovascular Quality and Outcomes*, *10*(9), p.e003800.

[37]. Pistoia Alliance, https://www.pistoiaalliance.org/, [Ziyaret Tarihi: 10 Mart 2019]

[38]. Dataflog, https://datafloq.com/read/blockchain-technology-use-cases-statistics-benefit/3719, [Ziyaret Tarihi: 10 Mart 2019]

[39]. Coindesk, https://www.coindesk.com/research/state-of-blockchain/2017/q2, [Ziyaret Tarihi: 10 Mart 2019]

[40]. Deloitte, https://www2.deloitte.com/content/dam/Deloitte/us/Documents/financial-services/us-dcfs-blockchain-in-cre-the-future-is-here.pdf, [Ziyaret Tarihi: 10 Mart 2019]

[41]. Antonopoulos, A.M., 2017, *Mastering bitcoin: Programming the open Blockchain*, " O'Reilly Media, Inc.".

[42]. Beck, R., 2018, Beyond bitcoin: The rise of blockchain world, *Computer*, *51*(2), pp.54-58.

[43]. Pilkington, M., 2016, Blockchain technology: principles and applications, In *Research handbook on digital transformations*. Edward Elgar Publishing.

[44]. Singh, S. and Singh, N., 2016, Blockchain: Future of financial and cyber security, In *2016 2nd international conference on contemporary computing and informatics (IC3I)* (pp. 463-467). IEEE.

[45]. Croman, K., Decker, C., Eyal, I., Gencer, A.E., Juels, A., Kosba, A., Miller, A., Saxena, P., Shi, E., Sirer, E.G. and Song, D., 2016, On scaling decentralized blockchains, In *International conference on financial cryptography and data security* (pp. 106-125). Springer, Berlin, Heidelberg.

[46]. Zyskind, G. and Nathan, O., 2015, Decentralizing privacy: Using blockchain to protect personal data, In *2015 IEEE Security and Privacy Workshops* (pp. 180-184). IEEE.

[47]. Kosba, A., Miller, A., Shi, E., Wen, Z. and Papamanthou, C., 2016, Hawk: The blockchain model of cryptography and privacy-preserving smart contracts, In *2016 IEEE symposium on security and privacy (SP)* (pp. 839-858). IEEE.

[48]. Meiklejohn, S. and Orlandi, C., 2015, Privacy-enhancing overlays in Bitcoin, In *International Conference on Financial Cryptography and Data Security* (pp. 127-141). Springer, Berlin, Heidelberg.

[49]. Wang, L., Shen, X., Li, J., Shao, J. and Yang, Y., 2019, Cryptographic primitives in blockchains, *Journal of Network and Computer Applications*, *127*, pp.43-58.

[50]. Gupta, S.S., 2017, *Blockchain*, John Wiley & Sons, Inc.

[51]. Handschuh, H., Helena, H. and Naccache, D., 2000, SHACAL.

[52]. Rogaway, P. and Shrimpton, T., 2004, Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance, In *International workshop on fast software encryption* (pp. 371-388). Springer, Berlin, Heidelberg.

[53]. Diffie, W. and Hellman, M., 1976, New directions in cryptography, *IEEE transactions on Information Theory*, *22*(6), pp.644-654.

[54]. Johnson, D., Menezes, A. and Vanstone, S., 2001, The elliptic curve digital signature algorithm (ECDSA), *International journal of information security*, 1(1), pp.36-63.

[55]. Lamport, L., Shostak, R. and Pease, M., 2019, The Byzantine generals problem, In *Concurrency: the Works of Leslie Lamport* (pp. 203-226).

[56]. Baliga, A., 2017, Understanding blockchain consensus models, *Persistent*, *2017*(4), pp.1-14.

[57]. Sankar, L.S., Sindhu, M. and Sethumadhavan, M., 2017, Survey of consensus protocols on blockchain applications, In *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)* (pp. 1-5). IEEE.

[58]. Zheng, Z., Xie, S., Dai, H.N., Chen, X. and Wang, H., 2018, Blockchain challenges and opportunities: A survey, *International Journal of Web and Grid Services*, *14*(4), pp.352-375.

[59]. Saleh, F., 2019, Blockchain without waste: Proof-of-stake, *SSRN 3183935*.

[60]. Copeland, C. and Zhong, H., 2016, Tangaroa: a byzantine fault tolerant raft.

[61]. Larimer, D., 2018, Delegated proof-of-stake consensus.

[62]. Ghosh, M., Richardson, M., Ford, B. and Jansen, R., 2014, *A TorPath to TorCoin: Proof-of-bandwidth altcoins for compensating relays*, Naval Research Lab Washıngton Dc.

[63]. HyperLedger Sawtooth, https://sawtooth.hyperledger.org/docs/core/nightly/0-8/introduction.html [Ziyaret tarihi: 01 Ocak 2020]

[64]. P. technologies, https://wiki.parity.io/Proof-of-Authority-Chains [Ziyaret tarihi: 15 Aralık 2019]

[65]. Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W. and Qijun, C., 2017, A review on consensus algorithm of Blockchain, In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 2567-2572). IEEE.

[66]. Nguyen, G.T. and Kim, K., 2018, A Survey about Consensus Algorithms Used in Blockchain, *Journal of Information processing systems*, *14*(1).

[67]. Hao, Y., Li, Y., Dong, X., Fang, L. and Chen, P., 2018, Performance analysis of consensus algorithm in private Blockchain, In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 280-285). IEEE.

[68]. Chalaemwongwan, N. and Kurutach, W., 2018, State of the art and challenges facing consensus protocols on Blockchain, In *2018 International Conference on Information Networking (ICOIN)* (pp. 957-962). IEEE.

[69]. Back, A., 2002, Hashcash-a denial of service counter-measure.

[70]. Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W. and Qijun, C., 2017, A review on consensus algorithm of Blockchain, In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 2567-2572). IEEE.

[71]. Košt'ál, K., Krupa, T., Gembec, M., Vereš, I., Ries, M. and Kotuliak, I., 2018, On Transition between PoW and PoS. In *2018 International Symposium ELMAR* (pp. 207-210). IEEE.

[72]. Garay, J., Kiayias, A. and Leonardos, N., 2015, The bitcoin backbone protocol: Analysis and applications, In *Annual International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 281-310). Springer, Berlin, Heidelberg.

[73]. Judmayer, A., Stifter, N., Krombholz, K. and Weippl, E., 2017, Blocks and chains: introduction to bitcoin, cryptocurrencies, and their consensus mechanisms, *Synthesis Lectures on Information Security, Privacy, & Trust*, *9*(1), pp.1-123.

[74]. Buterin, V., 2014, A next-generation smart contract and decentralized application platform, *white paper*, *3*(37).

[75]. Xu, X., Weber, I., Staples, M., Zhu, L., Bosch, J., Bass, L., Pautasso, C. and Rimba, P., 2017, A taxonomy of blockchain-based systems for architecture design, In *2017 IEEE International Conference on Software Architecture (ICSA)* (pp. 243-252). IEEE.

[76]. Russinovich, M., Costa, M., Kerner, M. and Moscibroda, T., Microsoft Technology Licensing LLC, 2018, *Transaction processing for consortium blockchain network*, U.S. Patent Application 15/638,213.

[77]. Buterin, V., 2015, On public and private blockchains, https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/ [Ziyaret tarihi: 15 Temmuz 2019]

[78]. Ethereum project, https://ethereum.org/ [Ziyaret tarihi: 18 Temmuz 2019]

[79]. Hyperledger Blockchain, https://www.hyperledger.org/ [Ziyaret tarihi: 18 Temmuz 2019]

[80]. Ripple Blockchain, https://www.ripple.com/ [Ziyaret tarihi: 18 Temmuz 2019]

[81]. Quorum Blockchain, https://www.goquorum.com/ [Ziyaret tarihi: 18 Temmuz 2019]

[82]. Corda | Open Source Blockchain Platform for Business, https://www.corda.net/ [Ziyaret tarihi: 18 Temmuz 2019]

[83]. Hildenbrandt, E., Saxena, M., Rodrigues, N., Zhu, X., Daian, P., Guth, D., Moore, B., Park, D., Zhang, Y., Stefanescu, A. and Rosu, G., 2018, Kevm: A complete formal semantics of the ethereum virtual machine, In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)* (pp. 204-217). IEEE.

[84]. Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y. and Muralidharan, S., 2018, Hyperledger fabric: a distributed operating system for permissioned blockchains, In *Proceedings of the Thirteenth EuroSys Conference* (pp. 1-15).

[85]. Androulaki, E., Cachin, C., De Caro, A., Kind, A. and Osborne, M., 2017, Cryptography and protocols in hyperledger fabric, In *Real-World Cryptography Conference.*

[86]. Pongnumkul, S., Siripanpornchana, C. and Thajchayapong, S., 2017, Performance analysis of private blockchain platforms in varying workloads, In *2017 26th International Conference on Computer Communication and Networks (ICCCN)* (pp. 1-6). IEEE.

[87]. Valenta, M. and Sandner, P., 2017, Comparison of ethereum, hyperledger fabric and çorda, *no. June*, pp.1-8.

[88]. Omohundro, S., 2014, Cryptocurrencies, smart contracts, and artificial intelligence, *AI matters*, *1*(2), pp.19-21.

[89]. Frustaci, M., Pace, P., Aloi, G. and Fortino, G., 2017, Evaluating critical security issues of the IoT world: Present and future challenges, *IEEE Internet of things journal*, 5(4), pp.2483-2495.

[90]. Weber, I., Xu, X., Riveret, R., Governatori, G., Ponomarev, A. and Mendling, J., 2016, Untrusted business process monitoring and execution using Blockchain, In *International Conference on Business Process Management* (pp. 329-347). Springer, Cham.

[91]. Khan, M.A. and Salah, K., 2018, IoT security: Review, blockchain solutions, and open challenges, *Future Generation Computer Systems*, *82*, pp.395-411.

[92]. Conti, M., Dehghantanha, A., Franke, K. and Watson, S., 2018, Internet of Things security and forensics: Challenges and opportunities.

[93]. Kolias, C., Kambourakis, G., Stavrou, A. and Voas, J., 2017, DDoS in the IoT: Mirai and other botnets, *Computer*, *50*(7), pp.80-84.

[94]. Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M., 2013, Internet of Things (IoT): A vision, architectural elements, and future directions, *Future generation computer systems*, *29*(7), pp.1645-1660.

[95]. Abomhara, M. and Køien, G.M., 2014, Security and privacy in the Internet of Things: Current status and open issues, In *2014 international conference on privacy and security in mobile systems (PRISMS)* (pp. 1-8). IEEE.

[96]. Sun, J., Yan, J. and Zhang, K.Z., 2016, Blockchain-based sharing services: What blockchain technology can contribute to smart cities, *Financial Innovation*, *2*(1), pp.1-9.

[97]. Samaniego, M. and Deters, R., 2016, Blockchain as a Service for IoT, In *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)* (pp. 433-436). IEEE.

[98]. Atzori, M., 2017, Blockchain-based architectures for the internet of things: A survey, *Available at SSRN 2846810*.

[99]. Reyna, A., Martín, C., Chen, J., Soler, E. and Díaz, M., 2018, On blockchain and its integration with IoT. Challenges and opportunities, *Future generation computer systems*, *88*, pp.173-190.

[100]. Jesus, E.F., Chicarino, V.R., de Albuquerque, C.V. and Rocha, A.A.D.A., 2018, A survey of how to use blockchain to secure internet of things and the stalker attack, *Security and Communication Networks*, *2018*.

[101]. Rouhani, S. and Deters, R., 2019, Security, performance, and applications of smart contracts: A systematic survey, *IEEE Access*, 7, pp.50759-50779.

[102]. Gong, L., 1989, A Secure Identity-Based Capability System, In *IEEE symposium on security and privacy* (pp. 56-63).

[103]. Malviya, H., 2016, How Blockchain will defend IOT, *Available at SSRN 2883711*.

[104]. Christidis, K. and Devetsikiotis, M., 2016, Blockchains and smart contracts for the internet of things, *Ieee Access*, *4*, pp.2292-2303.

[105]. Ouaddah, A., Abou Elkalam, A. and Ait Ouahman, A., 2016, FairAccess: a new Blockchain-based access control framework for the Internet of Things, *Security and Communication Networks*, *9*(18), pp.5943-5964.

[106]. Hammi, M.T., Hammi, B., Bellot, P. and Serhrouchni, A., 2018, Bubbles of Trust: A decentralized blockchain-based authentication system for IoT, *Computers & Security*, 78, pp.126-142.

[107]. Xia, Q., Sifah, E.B., Smahi, A., Amofa, S. and Zhang, X., 2017, BBDS: Blockchain-based data sharing for electronic medical records in cloud environments, *Information*, *8*(2), p.44.

[108]. Almadhoun, R., Kadadha, M., Alhemeiri, M., Alshehhi, M. and Salah, K., 2018, A user authentication scheme of IoT devices using blockchain-enabled fog nodes, In *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)* (pp. 1-8). IEEE.

[109]. Wikipedia, http://en.wikipedia.org/wiki/Authentication, [Ziyaret tarihi: 05 Ekim 2019].

[110]. Schneier, B., 2005, Two-factor authentication: too little, too late, *Communications of the ACM*, 48(4), p.136.

# ÖZGEÇMİŞ

| Kişisel Bilgiler | |
|---|---|
| Adı Soyadı | Mohammed ALSADI |
| Doğum Yeri | Filistin |
| Doğum Tarihi | 19.02.1987 |
| Uyruğu | ❒ T.C. ☑ Diğer: FİLİSTİN |
| Telefon | |
| E-Posta Adresi | mehmet.alsadi@gmail.com |
| Web Adresi | |

| Eğitim Bilgileri | |
|---|---|
| **Lisans** | |
| Üniversite | Arab American University |
| Fakülte | Information Technology |
| Bölümü | Computer Information Technology |
| Mezuniyet Yılı | 09.07.2009 |

| Yüksek Lisans | |
|---|---|
| Üniversite | İstanbul Üniversitesi |
| Enstitü Adı | Fen Bilimleri Enstitüsü |
| Anabilim Dalı | Enformatik Anabilim Dalı |
| Programı | Enformatik Programı |

| Doktora | |
|---|---|
| Üniversite | İstanbul Üniversitesi |
| Enstitü Adı | Fen Bilimleri Enstitüsü |
| Anabilim Dalı | Enformatik Anabilim Dalı |
| Programı | Enformatik Programı |

**Makale ve Bildiriler**

**Alsadi, M.,** Yıldırım, S., Gülseçen, S., Köse, B. Ö., & Coşkun, V. (2019, October). Akıllı Araç Ekosistemlerinde Blockchain Tabanlı Güvenli Veri Yönetim Modeli. In 2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (pp. 1-5). IEEE.

**Alsadi, M.,** Gülseçen, S., Sinan, K. A. R. A., Köse, B. Ö., & Coşkun, V. Blockchain Tabanlı Bir Veri Yönetim Modeli. Journal of Information Systems and Management Research, 1(1), 31-36.

Ok, K., Cevikbas, C., Coskun, V., **Alsadi, M.**, & Ozdenizci, B. (2016). Security Analysis of SIMSec Protocol. International Journal of Computer, Electrical,

Automation, Control and Information Engineering Vol:10, No:2.

**Alsadi, M.,** Mantar, H. A., Coskun, V., Ok, K., & Ozdenizci, B. (2016). Challenges and Risks of Developing a Payment Facilitator Model. J. Inf. Secur. Res., 7(3), 109-117

**Alsadi M.**, Karlidere T.,Ozdenizci B., Coskun V., Secure Element on the Cloud System, International Conference on Computer Science and Engineering, Tekirdag, TURKEY, 20-23 October 2016, pp. 550-556.

Celenlioglu, M. R., **Alsadi, M.**, & Mantar, H. A. (2015, July). Design, implementation and evaluation of SDN-based resource management model. In 2015 7th International Conference on New Technologies, Mobility and Security (NTMS) (pp. 1-5). IEEE. doi: 10.1109/NTMS.2015.7266484. Paris, France.

Ok K., Cevikbas R. C., **Alsadi M.**, Ozdenizci B, Coskun V., Implementation of a Key Exchange Protocol for Secure Communication between SIM Card And Service Provider, Academics World 15th International Conference, Bangkok, Tayland, 29 October 2015, pp. 35-37

Ozdenizci, B., Ok, K., **Alsadi, M.**, Coskun, V., & Soylemezgiller, F. (2014). DEVELOPMENT OF NFC ENABLED LOYALTY APPLICATION: Technical and Business Opportunities Academic Journal of Science (AJS), 2014, 3 (1), pp. 141-149. Rome, Italy.

Coskun V., Ozdenizci B., Ok K., **Alsadi, M.**, NFC loyal system on the cloud, 7th International Conference on Application of Information and Communication Technologies (AICT, 23-25 October 2013, pp. 1-5. ). Baku, AZERBAIJAN.

Ozdenizci, B., Alsadi, M., Ok, K., & Coskun, V. (2013). Classification of NFC Applications in Diverse Service Domains. International Journal of Computer and Communication Engineering, 2(5), 614. DOI: 10.7763/IJCCE.2013.V2.260, Thailand.

Coskun V., Ozdenizci B., Ok K., **Alsadi M.**, Soylemezgiller F., Design and Development of NFC Enabled Loyalty System, 6th International Conference of Advanced Computer Systems and Networks: Design and Application, 16-18 September 2013, pp. 42-45. Lviv, Ukraine.