BLOCKCHAIN BASED CONTEXT AWARE ACCESS CONTROL STRUCTURE IMPLEMENTATION FOR SECURITY OF INTERNET OF THINGS SYSTEM

A Thesis Submitted to
the Graduate School of
Izmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Computer Engineering

by
Aslı KUL

March 2022
İZMİR

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisors, Prof. Dr. Onur DEMİRÖRS and Prof. Dr. Yusuf Murat ERTEN, for their patience, excellent humanity and guidance throughout study.

I am proud and grateful for the time I work with my advisors.

Last but not the least, I would like to thank all my KUL family and all loved ones who have encouraged and supported me whenever I needed it.

ABSTRACT

BLOCKCHAIN BASED CONTEXT AWARE ACCESS CONTROL STRUCTURE IMPLEMENTATION FOR SECURITY OF INTERNET OF THINGS SYSTEM

Nowadays, Internet of Things (IoT) devices, which have started to be included in our lives with the developing technology, are one of the popular working topics. While they are a means of transporting many important data in terms of usage areas, IoT devices have brought security concerns due to their being a new technology and technical limitations. The use of standard access control algorithms is insufficient for IoT environments due to their complexity and dynamism. In this study, considering the importance of sensitive critical data carried by IoT environments, Context Aware IoT Rule Based Access Control Algorithm, which is a proposed algorithm to ensure the security of interaction with IoT environments, is aimed to be integrated and used to create a reliable IoT environment by taking advantage of the security promising Blockchain technology. It is due to the use of distributed and cryptography methods that are widely used today.

Keywords: IoT, Smart Contract, Access Control, Security, Blockchain, Context Aware, Internet of Things

ÖZET

NESNELERİN İNTERNETİ SİSTEMLERİNİN GÜVENLİĞİ İÇİN BLOK ZİNCİR TABANLI BAĞLAM DUYARLI ROL ERİŞİM DENETİM YAPISI UYGULAMASI

Günümüzde gelişen teknoloji ile birlikte hayatımıza dahil olmaya başlayan Nesnelerin İnterneti (IoT) cihazları popüler çalışma alanlarından biridir. Kullanım alanları itibariyle birçok önemli verinin taşınmasına aracı olurlarken henüz yeni bir teknoloji olması ve teknik kısıtlı yapılarından dolayı IoT cihazları güvenlik ile ilgili endişeleri de beraberinde getirmişlerdir. Standart erişim control algortimaları karmaşıklıkları ve dinamik olmamaları sebebyile IoT ortamları için yetersiz kalmaktadır. Bu çalışmada IoT cihazlarının taşıdığı hassas kritik verilerin ehemmiyeti de göz önünde bulundurularak cihazlar ile etkileşimin güvenilirliğini sağlamak için IoT ortamları için önerilen bir algoritma olan modelin yine günümüzde kullanım alanı yaygınlaşan dağıtık ve kriptografik yöntemler kullanımı sebebiyle güvenlik vadeden Blockhain teknolojisinin avantajlarından faydalanarak güvenilir bir IoT ortamı oluşturulmak amaçlanmıştır.

Anahtar Kelimeler: IoT, Akıllı Sözleşmeler, Erişim Kontrolleri, Güvenlik, Blockchain Bağlam Duyarlı, Nesnelerin İnterneti

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Table **Page**

LIST OF ABBREVIATIONS

Abbreviations

ABAC	Attribute Based Access Control
AAT	Attribute Assignment Table
CA-RBAC	Context-Aware Role Based Access Control
CA-IRBAC	Context-Aware IoT Role Based Access Control
CDB	Context Database
CR	Context Rule
IoT	Internet of Things
MCR	Multiple Context Rules
MySQL	My Structured Query Language
OA	Object Attribute
OAOA	Object Attribute Operation Assignment
PEP	Policy Enforcement Point
PDP	Policy Decision Point
PCM	Policy Configuration Manager
RBAC	Role Based Access Control
SA	Subject Attribute
SAOA	Subject Attribute Operation Assignment
UCON	Usage Control Model

CHAPTER 1

INTRODUCTION

The adventure of digital communication, which started with the communication between two homeowners in the early 1960s, developed with the emergence of the internet and the connection of many computers in the 90s, and this adventure continued with the addition of mobile devices to the World Wide Web. Afterwards, host, web, mobile and human communication got involved in this adventure and continues to evolve with the Internet of Things, a technology that does not require human intervention in the mid-2010s. The Internet of Things (IoT) is a system of interrelated devices capable of transmitting data over a network using unique identifiers (UIDs) without the need for any outside intervention [1].

As the usage areas of IoT devices that integrate the physical world and the virtual world via the Internet increase day by day, important data collected in different areas also increases. This number, the amount of data produced by IoT devices, is expected to reach 73.1 ZB (zettabytes) by 2025[2]. IoT data, which can be a source for many different areas and enable organizations to make smarter decisions and get faster results, makes it important to meet security requirements at many points, from the transmission of important data to its storage and sharing between different services. While it is estimated that the number of IoT devices connected to the Internet will exceed 25.4 billion in 2030, another forecast is that the number of devices connected to the Internet per minute will be 152,200 by 2025.

While technological innovations provide benefits in many ways, they also cause different security vulnerabilities. The fact that IoT devices are lightweight and produced by different manufacturers causes hardware-based security to be insufficient [3]. Considering the usage areas, the use of IoT devices in environments that produce sensitive personal data such as smart homes and hospitals gains importance in terms of controlling access to the produced data.

Access control can be defined as accessing an object only by authorized parties. Various methodologies are used to ensure data security, from role-based access control methods to attribute-based access control methods. Standard access control algorithms are complex for IoT environments and not very compatible with their dynamic nature. It is important that the access control algorithm to be used for authorization and permission control in IoT environments is compatible with its dynamic structure and flexibility, taking into account the limitations of IoT devices.

Another important security requirement in data security is that the data must be immutable. The fact that the necessary permissions and authorizations defined for access control methods and algorithms cannot be changed is of vital importance in order not to cause security weakness. In the context of this security need, Blockchain Technology, which is another research area today, emerges as a distributed database where encrypted hashed transactions can be tracked. [4] Constructed as a chain of an immutable block structure linked to each other [5], each block contains the hash code of the previous block structure in its own block on the chain it belongs to block [6] Blockchain, which is a distributed system, provides a significant advantage in terms of security by providing us with an immutable structure, since the change of any block must be approved by the majority. On the other hand digital smart contracts running on the blockchain also allow us to make immutable contracts that run based on computer protocols, and thus it is used for security purposes in different areas such as the Internet of Things (IoT), smart cities, and the protection of personal data.

In this thesis, Context Aware Access Control Algorithm (CA-RBAC), an awareness access control algorithm which is proposed to provide dynamic access control in IoT environments is proposed. It will be used by simulating Smart Contracts integration with a smart patient tracking system scenario and its performance will be presented.

1.1. Motivation and Aim

Whichever access methodology is used, the security of the environment in which rights, roles and permissions are defined, stored and evaluated is vital. Even in a well-designed access control algorithm, changing the defined privileges by unauthorized parties and evaluation of this algorithm will be a weak point in the access control mechanism. In this project it is aimed to use blockchain systems, which are based on cryptographic methods for this purpose. This technology is increasingly preferred for secure systems because of their unchangeable feature, and is the subject of research and application for access control authorization. With the use of the operation-based, flexible, and easy-to-manage CA-RBAC model, it has been tried to demonstrate its usability in the field of security, which is an important requirement for IoT solutions.

The restriction of IoT environments and the contribution of the used technologies and the algorithms are given on the table.

Table 1.1.1. IoT Restriction and the Contribution of the Used Technologies

Features and Restrictions		Contributions		
IoT	**Standart Access Control Algorithms**	**Blockchain**	**Smart Contract**	**Context Aware Access Control Algorithm**
Centralized		Decentralized	Autonomy	Dynamic
Limited Bandwidth and Resource	Role, rule numbers etc. increase cost for IoT	Cost Saving	Trust	Flexible
Large Number of Devices	Complex for IoT	Immutability	Security	Reduce Complexity
Security Problem		Security	Transparency	
Scalability is low and expensive	Not flexible	Scalable		

1.2. Thesis Outline

The thesis is organized in five main sections as follows. In Chapter 2, general information and background details are given. IoT Devices’ basic architecture, main security requirements and most important challenges for the development of secure environments are given. The Blockchain and why it is utilized and Smart Contract Structure and usage areas for IoT application in general are also mentioned in this section. In the Chapter 3, other studies on access control algorithms are briefly mentioned. Chapter 4 includes the design and implementation stages of the application. While 4.1 describes the algorithm, 4.2 includes the system design to be implemented and 4.3 contains brief information about the technologies used. The scenario and implementation details of the system simulation are explained in sections 4.4 and 4.6. The results of the study and comparisons of different algorithms are given in the Chapter 5.

CHAPTER 2

BACKGROUND

2.1. IoT Environment

Internet of Things is a concept that was first used by Kevin Ashton in a presentation in 1991. It refers to connected devices which can exchange data and change their behavior according to the context information of environment thanks to the Internet.

The IoT environment, which consists of devices connected to each other and in continuous data communication, is expected to reach worldwide from 8.74 million in 2020 to more than 25.4 billion in 2030 [7], showing how large the IoT environment is.

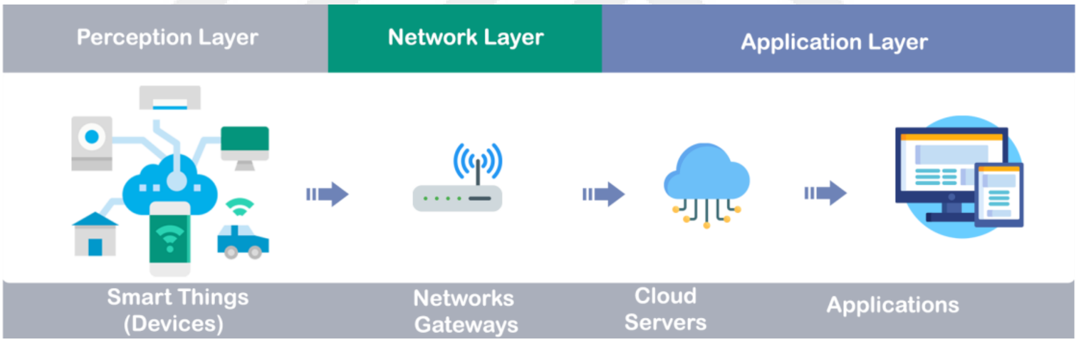


Figure 2.1.1. Layers of IoT Application

With IoT technology, many data such as temperature, light, pressure, image obtained from sensor devices are accessible via wireless communication techniques such as Bluetooth, Infrared, Zigbee, Wifi, thus enabling the establishment of controllable and decision-making smart systems.

The main usage areas of IoT usage can be grouped as follows:

- ✓ **Smart Home Automations:** Smart home and building systems that can make decisions according to safe and remote monitoring and controllable environment conditions.
- ✓ **Industrial:** Production tracking systems that require real-time monitoring to increase safety and reliability

- ✓ **Transportation:** Unmanned autonomous vehicle, traffic monitoring, emergency rescue, smart parking and traffic light systems.
- ✓ **Environment:** Metrology monitoring, emergency early warning systems
- ✓ Agriculture: Intelligent agriculture, irrigation systems
- ✓ **Health:** remote monitoring of real-time or periodic health information (blood sugar, blood pressure, heartbeat, body temperature, number of steps, instant physical condition, etc.) of people with chronic diseases or elderly people in need of care.
- ✓ In this ever-growing network with billions of devices connected, security emerges as a major concern.

2.1.1. Challenges

Internet of Things devices create a source for accessing sensitive data that is generated in many areas of daily life. Due to usage areas of IoT devices, they receive a lot of sensitive data, and due to their limited lightweight nature, they are insufficient in the security solutions that can be applied to ensure the confidentiality of the sensitive data they carry.

One of the main challenges to creating a secure IoT environment is that IoT devices have limited resources and cannot handle operations that require extensive computation. On the other hand, the current centralized IoT architecture can cause the entire system to be inaccessible because it can be managed from a single point due to denial-of-service attacks.

In addition to physical devices, IoT systems also contain many endpoints that need security on their own, such as mobile applications, APIs, network interfaces database. Vulnerability at any point in the system poses a threat to the entire system. Therefore, it is important that access to any part of the system is reliable and authorized.

For these reasons, there is a need for both Access control to be both scalable and decentralized to handle the exponentially growing IoT devices. Due to this need, a solution was sought for the security of IoT devices by using a decentralized Blockchain structure together with a dynamic and context-based flexible access control structure suitable for IoT this architecture.

2.2. Blockchain

The Blockchain technology, which is one of the popular working areas; stands out as an alternative solution to ensure security in the IoT environment in addition to methods such as the use of different encryption methods to provide reliable IoT systems, the development of reliable protocols suitable for the IoT structure. Blockchain-based systems are distributed data structures consisting of a combination of cryptographic, public key infrastructure, implementation of peer-to-peer networking and decentralized consensus [8].

Consensus and immutability are at the forefront of the basic concepts of security related to Blockchain. Transactions that have already been confirmed cannot be changed and records are stored at all decentralized points, while also using heavy cryptography. Each block in the Blockchain is hashed and linked with the previous block, and a change in a block requires the hash to be changed, playing an important role in ensuring security and immutability.

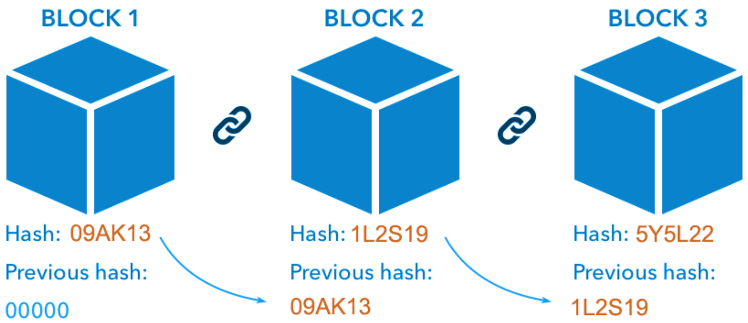


Figure 2.2.1. Block Structure in Blockchain

When the distributed architecture, in which data is kept at many points, is compared, the advantages and disadvantages of this architecture can be explained as follows:

Although the fault tolerance in centralized architectures depends on a single point, it becomes somewhat more stable in decentralized architectures, but in distributed architectures, errors do not stop the system from working, which makes it the most reliable systems in terms of stability and security.

System growth is unlimited in distributed architectures and does not have a negative impact on system performance, the only disadvantage is that it is more costly to maintain. In addition, the fact that it is easy to develop due to addable nodes explains the reason for the proposal to use Blockchain technology with a distributed architecture.

Table 2.2.1. Comparison of the centralized and decentralized model [9]

	Blockchain System	**Server Client System**
Network Type	Decentralized	Centralized
Features of network	High bandwidth and low expensive system maintenance cost	Easy implementation
Network problem	Other peers are still connectable	The whole system crashes with single point failure
Reliability	Cannot modified by outside reliable	Less reliable because review is open

2.3. Smart Contract

A smart contract is a transaction protocol that satisfies the need for trusted agents, created to automatically execute, control or document relevant events and actions according to predetermined terms. The term smart contracts were first proposed by Nick Szabo in 1994 and started to be implemented with bitcoin technology in 2008, as the digital platform and computing power to support this proposed system increased with technology.

We can define the operation of smart contracts with the following 4 steps.

✓ *Pre-Defined Contracts:* The parties agree on the policy to be applied. And it is hard coded on the smart contract.

✓ *Events:* It is ensured that the policy for the smart contract agreed with the events is triggered.

✓ Execution: Pre-defined terms policy is executed by smart contracts.

✓ *Settlement:* The transaction is settled, and details are recorded on the Blockchain.

It speeds up the process as it eliminates the need for intermediary institutions and organizations. Since it is kept encrypted and distributed in the Blockchain, it also has the security features of the Blockchain.

2.4. Access Control Algorithms

It is vital to take security measures and ensure confidentiality, integrity and availability to protect information-based systems that process data from unauthorized access or disruption of data integrity. One of the basic requirements of a secure system is to apply an accurate and sufficient access control model. It is necessary to authorize the system resources access according to the predefined security policy.

The Rule Based Access Control (RBAC) and Attribute Based Access Control (ABAC)are two types of access control methods based on the access control method applied in the thesis. In RBAC, users cannot access resources directly. Instead, they can access depending on their role. ABAC, on the other hand, considers user, resource, and environment properties to provide access rights. Their general structure and features are discussed in the next subsection.

2.4.1. RBAC: Role Based Access Control

Role-based access control (RBAC) allows access only to the information needed, while denying access to additional fields that are not relevant. A role determines the permissions granted to it and ensures that lower-level sensitive information cannot be accessed, or higher-level tasks cannot be performed.

While it is easy to manage a particular user's permissions, the increasing number of roles that need to be created in large organizations causes the need for system storage resources and processing power to increase. At the same time, managing too many roles becomes one of the main problems.

2.4.2. ABAC: Attribute Based Access Control

Attribute Based Access Control defines an access control paradigm whereby access rights are granted to users using policies which combine attributes together. [7]

ABAC is based on qualifications. These can be user attributes, attributes of the application to be accessed, and attributes of environmental conditions and are known as policy-based access control. ABAC offers a more flexible controllable access model, considering all defined attributes. The main limitation is that it provides flexibility, provides detailed access control, as well as the difficulty of its management and the creation of a security policy.

2.4.3. CA - IRBAC: Context Aware – IoT Rule Based Access Control

Since the IoT environment consists of too many variable components and environmental context, the access control models mentioned in sections 2.4.1 and 2.4.2 are insufficient for the IoT environment although the management of the person and his/her trace in role-based access control. Context Aware – IoT Rule Based Access Control (CA-IRBAC) proposed to provide access control for IoT environments is an access control model consisting of the combination of CA-RBAC and ABAC, which is put forward to transfer the easy management features of RBAC to ABAC [8].

Through CA - IRBAC model, which is proposed to overcome the difficulty of assigning a role to each subject, it is aimed to overcome the role-explosion problem by grouping the limited number of operations that can be done in the system according to the subjects. At the same time, with this model a more flexible structure was created by associating each subject with the attributes of the subjects instead of associating them with operations.

Table 2.4.3.1. Access Control Algorithms Comparison for IoT Environments

	RBAC	ABAC	CA-IRBAC
Method	Role based	Attribute Based	Context aware – rule based
Management	Role managing is a problem for huge organizations	Management is difficult	It overcomes the role explosion problem.
Flexibility	Roles, users and operations are tightly bound, not flexible	Flexible because of attributes.	Flexible because of attributes and associating with them instead of operations.

CHAPTER 3

RELATED WORKS

In this study, the main goal is to make access control more flexible, reliable and secure for the IoT environment. In order to achieve this goal, the CA-IRBAC, is proposed as an alternative to the existing RBAC and ABAC access controls models in terms of its dynamic structure and reducing complexity. Combining RBAC and ABAC, it is expected to create a stronger system against attacks by providing extra reliability which is further enhanced by combining the CA-IRBAC algorithm with the Smart Contract infrastructure.

One of the works done by including contexts in access controls belongs to the following study. (Covington et al., 2001) has done a study in which they expanded the structure of the rule-based algorithm by adding the security factors captured in the environment as a context. In the study, permissions are associated with subject and environmental factors. The decision mechanism works depending on whether the environmental factors are within the specified ranges for the defined people. Environmental factors are also part of the roles in this study, which is not desirable as it means too many components for the IoT domain.

(Bai et al., 2014) is a security application called ConUCON, which uses contexts to provide security and privacy in access control, in another similar study developed for web of Things applications that are more similar to IoT environments and the main difference is that objects are grouped and managed separately instead of users.

(Abdella et al., 2016) also made a study for mobile applications by matching the context with the Rule based access control model and its permissions features. In order for objects to be allowed to access, there is a conditionally defined context information. Despite the combined use, an efficient dynamic access control could not be observed in this study.

In another project (Long Xu and Yang Li,2020), a novel Capability Based Access Control Model is proposed (NCBAC) against the risks caused by the centralized access control algorithms. By defining role and attribute sets for smart contracts, it is ensured to benefit from the advantages of Capability-based access control's decision-making mechanism. The goal is to provide a more flexible, expandable, and high granularity access control. In this study, the system parts are: IoT devices, storage devices, Blockchain Management Center, Smart Contract issuer. In this system, there is an IOT system abstracted from other sources. All information required during the access control process, such as user, administrator, role attribute permissions, are kept on the smart contract. This is the main difference from our proposed system.

Another project is developed by Priyanka Kamboj1 & Shivang Khare1 & Sujata Pal User (2021). In this study, role-based access control is implemented using Blockchain. In this way, it is aimed to provide a decentralized access control. Using the Ethereum platform, the assignment of role and user mappings was done on the Blockcain, and a reliable system was tried to be established against men in the middle attacks. The access structure is based on 3 pillars in this project: users, roles and permissions. and these 3 bases match each other as user role mapping role permission mapping and roles user mapping. It was emphasized that the main problem in the project was the authentication of the users, and it was emphasized that the users to whom the role and permissions would be assigned should be authenticated users in order to try to establish a reliable authentication mechanism. For the solution of the problem, the Blockchain structure has been presented as a necessity. In this system, the smart contract is mainly used for the following purposes: assigning roles to users, providing information to role assigners. A structure has been created that allows Smart Contract all necessary information to be kept, to define user, to keep user resource access permissions, to add and remove operations.

Leepakshi Bindra, Kalvin Eng, Omid Ardakanian and Eleni Stroulia proposed a system for smart buildings. In this project also Smart Contracts are also used and it is desired to design a reliable control system for access for the building. For this purpose, a person-based time and space-defined model is used instead of role rule or attribution-based access. The proposed model is based on the location that if people are authorized for a certain area, they can have access to all resources there. There are 3 basic structures in the system: location equipment and points. The person authorized for a certain location is also authorized on the equipment in that location and in other regions that must pass to reach the location. Since the access permissions depend on the locations, it can be said that the permissions are matched with the paths, which increases the time and cost for the first installation. Since there is no user-specific access restriction for different role levels for people or access to objects, it is another matter to consider whether this will create a weakness on security.

Didem Genç, et. al [12] focused on the inadequacy of access control algorithms used in IoT applications for IoT environment requirements in the project they worked on “Context-Aware Operation-Based Access Control for Internet of Things Applications”. They argue that current security policies do not consider context information in IoT environments, so current security policies are insufficient to catch malicious attempts. Highlighting the need for a new approach to access control, they argue that incorporating context information into the access method will make it a more dynamic, granular and easily manageable access method. In this context, they have developed a context sensitive access algorithm for IoT applications, which they foresee to eliminate these shortcomings. They blended the rule-based access method and the rule- and attribute-based logic of the attribute-based access algorithm with context information in IoT environments. Instead of matching roles and operations in the algorithm they created, they combined the object attributes in the IoT environment with the rules, resulting in a more dynamic and weak granular algorithm and a more applicable algorithm for IoT environments.

And finally, the aim of this thesis is to simulate the Context-Based IoT rule Based Access Control [12] algorithm, which is proposed as an access control algorithm suitable for the structure of IoT applications, and at the same time, the algorithm process is operated over the smart contract so access, permission, authorization, and control are secured with this way. It is aimed to provide a reliable access control, both suitable for the IoT structure and by making use of the Blockchain technology. In terms of scope, this project differs from other studies by the algorithm it uses and provides a more appropriate access control for the IoT applications. The study showed that the simulation of CA-IRBAC algorithm is an alternative solution for IoT applications to the lack of dynamism in standard algorithms and the difficult-to-manage structure of access control management. At the same time, by ensuring that the access control process is carried out through Smart contracts, a security weakness that may occur at this stage is prevented and the immutability of the access control process is ensured by the immutable structure of Blockchain technology.

Table 3.1. Comparison of Related Works

Project	Access Conrol Alg.	Context	Smart Cont.	Dynamic	Flex-ibility	Granu-larity	Security	Cost	Suitable for IoT
Covington et al., 2001	Rule Based AC	✓ Environment - context	-	-			✓	High (Rule set)	- Rules cost High
Bai et al., 2014		✓	-	✓			✓	Medium Context	Similer to IoT
Abdella et al., 2016	Rule Based AC	✓ Permissons - context	-	- Combination of context is complex			Low	High (Rule set)	not dynamic
Long Xu & Yang Li, 2020	Capability Based AC		✓	✓	✓	✓	✓ Men in the Middle		✓
Priyana Kamboj 1 & Shivang Khare1 &Sujata Pal1	Role Based	-	✓	✓	✓		✓ Men in the Middle	Medium	✓
Leepakshi Bindra.. 2021	Location time based	-	✓	✓	✓	-	- Smart Contract Source restriction problem	Location path defining is too complex	✓
Didem Genç, 2018	Context Aware-IoT rule Based Algortihm	Context Based	-	✓	✓ Context based and rule associate with attributes		Secure by algorithm but not enough for execution	Low	✓
Proposed CA-IRBAC with Smart Contract Solution	Context Aware-IoT Rule Based Algortihm	Context Based	✓	✓	✓ Context based and rule associate with attributes		(Smart Contract) Access control in safe	Low	✓

CHAPTER 4

PROPOSED DESIGN AND IMPLEMENTATION

The diversity of objects and users in the IOT environment does not exactly make the use of current common access control models very efficient. Instead, the use of the Context Aware IoT Rule Based Access Control method suggested in (Genc, D.,2018) studies [12] coincides with the dynamic nature of IoT environment. In addition to the proposed access control mechanism, it is aimed to increase the security level by using the smart contract structure and making the access control in a secure and unchangeable environment, ensuring that the access decision mechanism is closed to unauthorized changes.

With the use of recommended CA-IRBAC algorithm, while the complexity of other access control models is reduced and a flexible structure is created, smart contracts are also included in the system design, and access control is aimed to be carried out in a more secure and unchangeable environment.

4.1. CA-IRBAC Algorithm

In this model, which is faithful to the original, the general terms are defined as follows.

1. Operations (Op): Operations define actions that can be performed on objects for certain subjects.

2. Subjects (S): Defines entities that have access rights to objects for defined situations.

3. Attributes: Attributes are divided into 2 groups according to Subject attributes (SA) and Object attributes (OA).

4. Context Expression (Con) : Contexts are defined in the database as belonging to objects, subjects and environment.

5. Context Rule (CR) : Context rules are expressions that decide whether or not to allow requests to access objects.

The relation of operation and object attribute in the access algorithm to be implemented in this thesis is shown in the Figure 4.1.1

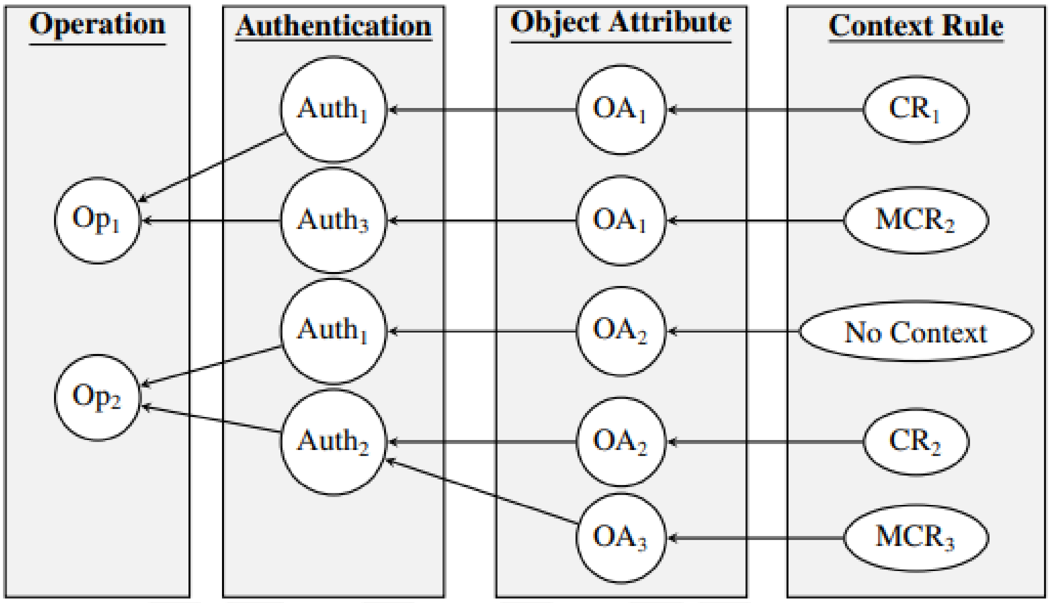


Figure 4.1.1. Operation and Object relation.

In the proposed model, each operation can be accessed with more than one authentication method. According to the evaluation results, the attributes of the objects that will be allowed or denied access are also associated with more than one authentication, and the operation object relationship is provided.

Allowing more than one authentication method to be used in the sampled method ensures that the access control is flexible. Considering the IoT environment, the existence of different devices and the diversity of end users who may want to access these devices also increase the diversity of authentication methods. While biometric authentication may be required for the activation of some devices, the authentication of a user who wants to remotely control a system designed for the smart system diversifies the need for different authentication methods.

The proposed algorithm in general is described as follows:

Table 4.1.1. Algorithm of CA-IRBAC

Algorithm 1

```
Input: AccessRequest (Rq<Sid, Oid, Op, Authenticated>)
Output: Access (Allow / Deny)
Operations ← List_Operations
if (Op) in list of Operations then
    SA ← get_attributes of requested subject (Sid)
else
    output → Access = Deny
end if
SA_List ← List_Subject_Attibutes (Op)
if (SA) in list of SA_ List then
    OA ← get_attributes of requested object (Oid)
else
    output → Access = Deny
end if
OA_List ← List_Object_Attibutes (Authenticated, Op)
if ((OA) in list of OA_ List) then
    Context ← get_context (Authenticated, Op, OA)
else
    output → Access = Deny
end if
Result ← evaluate (Context)
if Result = False then
    output → Access = Deny
    return(RequestDenied)
else
    output → Access = Grant
    return(RequestGranted)
end if
```

4. 2. System Design

In this study for proposed Access control model implementation IoT environment is decided as a smart home and patient monitoring, examination system.

There are certain objects that are relatively unchangeable compared to other components in the designed smart system or any IoT environment, and it is desired that these objects continue to work according to the rules defined from the beginning and that the access to these devices is carried out smoothly according to the determined rules. It is aimed to create a reliable access control mechanism that cannot be changed by creating a smart contract for each object in order to ensure access control, since it is the main goal that the objects work according to a certain policy and that their accessibility is reliable due to the data they carry.

The flow of designed model in general is listed as below.

- ✓ All defined operations, objects and subject variables to be used for access control in the system are recorded in the database.
- ✓ The smart contract is created for each object registered in the system.
- ✓ The smart contract is added to the Blockchain to be used to return the result as access or denial by taking the necessary parameters in access control.
- ✓ All objects with smart contracts that can be controlled through the application are displayed on the screen.
- ✓ The user chooses the operation for the object he wants to control.
- ✓ By taking the selected operation parameter, it is checked whether the subject has the authority for that operation and then the attributes of the object to be processed are checked.
- ✓ The necessary context is found according to the values of the object subject and transaction attributes and sent to the Smart Contract created for the relevant object for evaluation.
- ✓ By evaluating the context control with Smart Contract, access permission result is returned as acceptance or rejection all parameters that request access to the smart contract, such as subject, environment variables, and context evaluation, are recorded as a state in the Blockchain.

In this system IoT objects can be controlled either through the mobile application authentication or by biometric authentication for the smart home scenario. For the other scenarios which are detailed section 4.4.2 and 4.4.3 used authentication options are web application or web/mobile authentication.

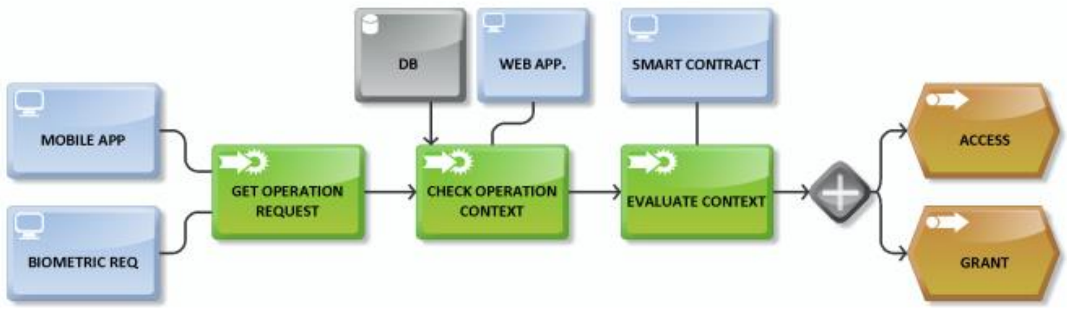


Figure 4.2.1. System Design

Whether the desired operation meets the requirements is checked over the web application according to the definitions in the database.

In this scenario all operations, subjects, objects, and their related attributes are defined in databases.

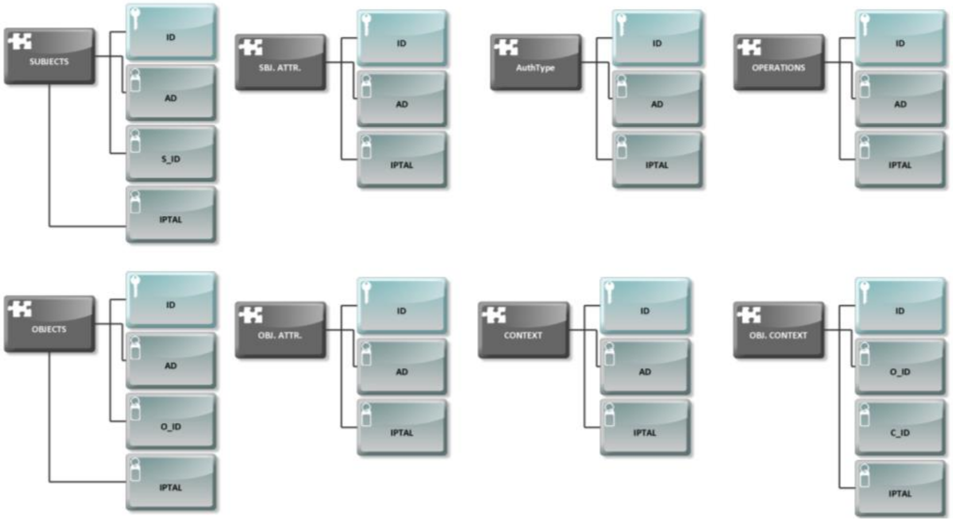


Figure 4.2.2. Database Design

If the operation's subject and object attributes are suitable for the requesting user (subject), the necessary parameters (attributes of subject/objects, context and relation) are taken from the database then sent to the smart contract and context rule evaluated via Smart Contract.

The result obtained from the evaluation of each object with the smart contract is obtained through the Web3.js interface and the access status result is returned, and the access is granted or denied.

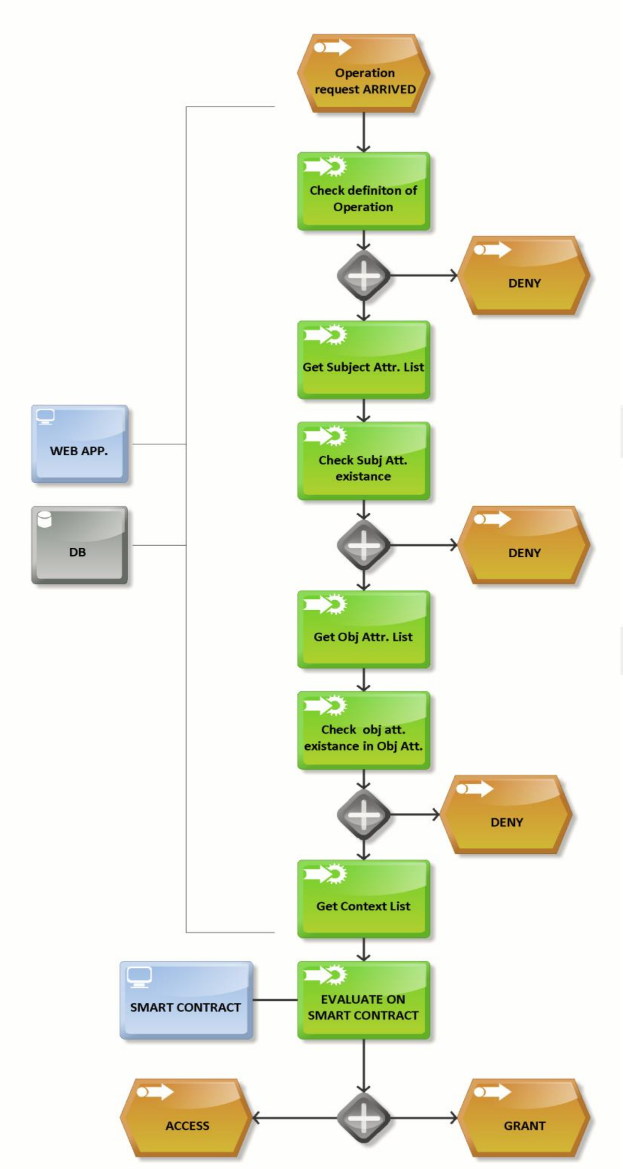

Operation request ARRIVED
Check definiton of Operation
DENY
Get Subject Attr. List
WEB APP.
Check Subj Att. existance
DB
DENY
Get Obj Attr. List
Check obj att. existance in Obj Att.
DENY
Get Context List
SMART CONTRACT
EVALUATE ON SMART CONTRACT
ACCESS
GRANT

Figure 4.2.3. Algorithm Flow Chart

The algorithm flow is defined as follows for Context Aware IoT Rule Based Access Control.

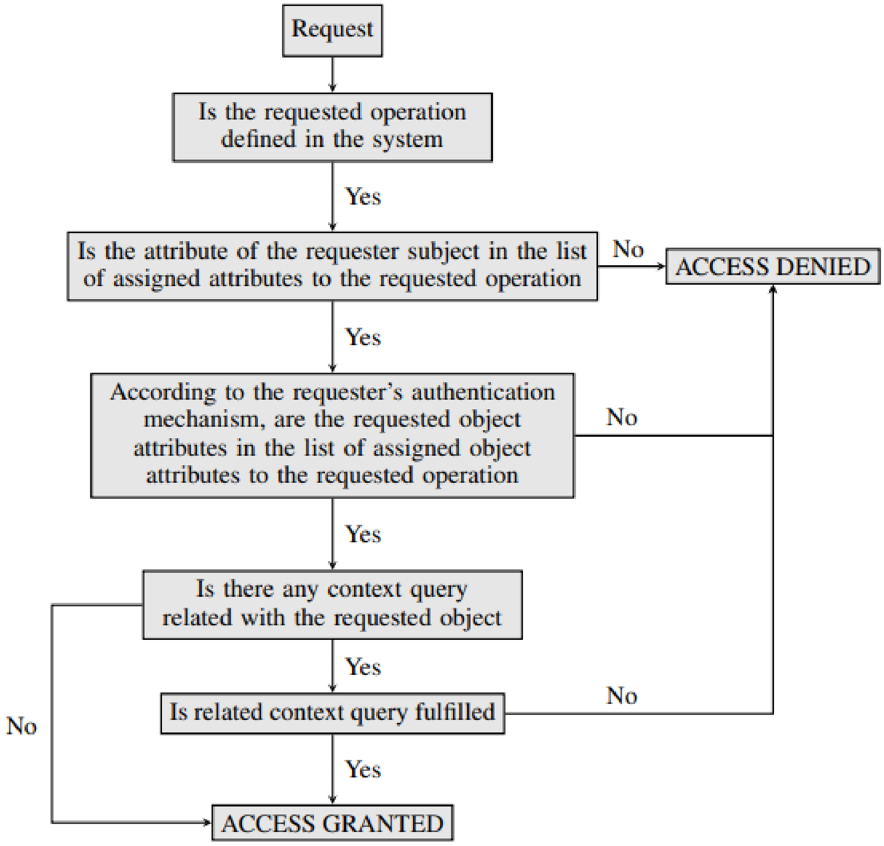

Request
Is the requested operation defined in the system
Yes
Is the attribute of the requester subject in the list of assigned attributes to the requested operation
No
ACCESS DENIED
Yes
According to the requester's authentication mechanism, are the requested object attributes in the list of assigned object attributes to the requested operation
No
Yes
Is there any context query related with the requested object
Yes
Is related context query fulfilled
No
No
Yes
ACCESS GRANTED

Figure 4.2.4. Flow Diagram of Working Principle

4.3. Used Technology

In this project, it is aimed to control the access of IOT devices in a remote patient monitoring system by authenticated system users by adapting CA-IRBAC, a proposed new access control algorithm, for use with smart contracts running on the Blockchain and keeping data in it.

Smart Contract codes are written using Solidity language. For the implementation of the algorithm that was decided to use in the thesis, the use of Truffle was preferred as the development environment.

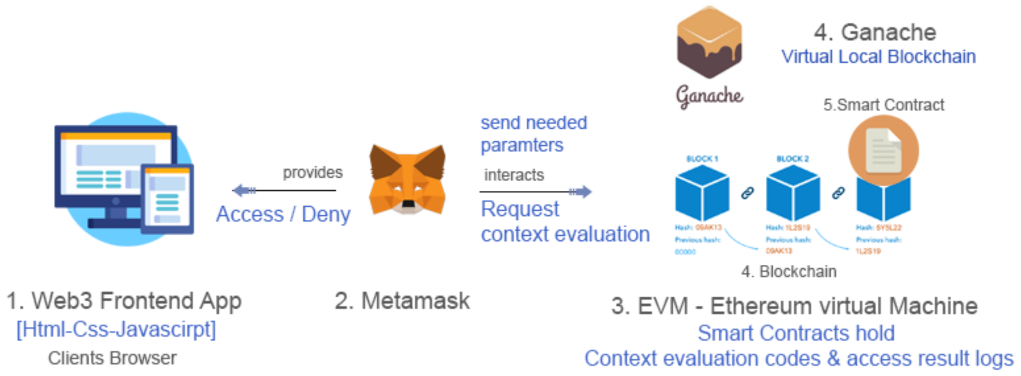


Figure 4.3.1. System Design

Thanks to Truffle, Smart Contracts written with Solidity were compiled, tested and distributed. The Ganache program was used to establish an Ethereum Blockchain locally to test Solidity smart contracts. The Metamask interacts between Ganache Ethereum Blockchain and web3.js client side.

4.3.1. Ethereum Blockchain

Ethereum is a platform for building distributed Blockchain applications where smart contracts can be created. Ethereum, which has a decentralized and distributed ledger; works on the Blockchain. Transactions are irreversible and visible to everyone because they are based on the principle of verifying and recording with hash functions.

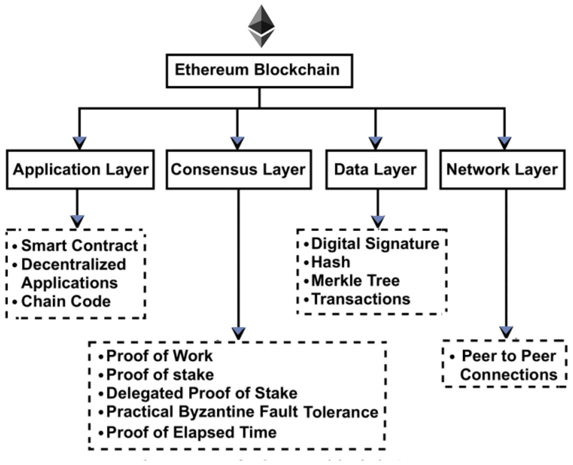


Figure 4.3.1.1. Ethereum Blockchain Layered Architecture

All Ethereum network participants have a copy of all transaction details made on the network. In Ethereum, where the Ether currency is used, the development tools are installed on the computer, allowing the application to be created and run on the main network.

An Ethereum Smart Contract account consists of the executable code, contract address, private storage space from the status bar, and balance.

4.3.2. Solidity

Solidity is an object-oriented, high-level language for implementing smart contracts. The Solidity is used same principles and syntax with JavaScript. It is highly preferred because it can be compiled on the lowest level machine, as well as allowing high-level code writing for the popular cryptocurrency Ethereum Blockchain network.

4.3.3. Php - Mysql

User operation interactions are simulated on web panel therefore Xamp 8.0.12 that is PHP development environment includes Apache server Php and MySql Database was installed. Requested operation parameters, attributes and situation are checked then sent to Smart contract to evaluate context related with requested operation according to CA-IRBAC Algorithm.

4.4. Scenarios

Within the scope of the project, three different scenarios are described, one of which is the smart home scenario proposed in the thesis named CA-IRBAC:

1. Smart Home
2. Smart patient observation room
3. Remote consulting appointment system non IoT daily life scenario.

First “Smart Home” scenario implementation, system variables and features are considered the same as in the first scenario. Implementation steps are given only for first scenario.

4.4.1. Scenario 1: Smart Home

The scenario in the thesis has been tried to be implemented by using the scenario of the proposed CA- IRBAC access control algorithm using smart contracts. The details of the created scenario components and features in the referenced thesis are as follows.

Subjects: Mother, Father, Child, Babysitter, Smart Home Application, Smart Healthcare Application

Subject Attributes: Title: Parent, Children, Babysitter, Healthcare App, Home App

Objects: Smart Door, Oven, Washing Machine, Camera, Wearable Blood Pressure

Object Attributes:

Object Type: Smart Door, Camera, Household Appliances, Wearable Devices

Environmental Attributes:

✓ Authentication: Mobile Device / Biometric

✓ Urgency: Urgent / not Urgent

✓ Time: School hours for children / Working hours for other system user

✓ Location: Smart home indoor / outdoor

✓ Parent’s Approval: allow / deny

✓ Door person control: yes / no

Object type and title are the defined attributes for objects and subjects respectively.

Smart Door: The smart can be opened according to predefined authentication methods and related situations in the system.

• The mother and father have the authority to open the door using biometric authentication in all cases.

• Mother and father can also can control it from a maximum distance of 10 meters from the house, using their mobile devices outside of working hours.

• While children can open the door from outside the home with the biometric verification method, at least one parent must be present in order to open the door while inside the home.

• The babysitter can only open the door from the outside with biometric verification during working hours. While the babysitter is at home, parental consent is required to open the door to the person outside. The babysitter has the authority to open the door since there is no one at the door.

• The smart home application is authorized to send the request to open the door when an emergency occurs and there is an ambulance less than 10 meters from the house.

Household Appliances:

• Parents are also authorized to turn on smart devices on their phones when they are out of the house.

• Children are not authorized to turn any smart device on or off.

• The babysitter can control smart devices through her/his mobile device during working hours and at home.

• Parents and the smart home application can turn off smart devices 30 minutes after they are turned on if there is no parent or babysitter at home.

Camera:

• The camera can only be viewed by parents with biometric authentication.

• Only in emergencies, parents can access the camera image via their mobile phone.

• Only in emergencies can the smart home application access the camera.

Wearable Blood Pressure:

• The smart health app always has permission to read data from the device.

• The smart health app can also read data in emergency situations for context verification.

4.4.2. Scenario 2 : Smart Health Application :

In this scenario, the smart patient monitoring room is the simulation environment.

Subjects: Patient, Responsible doctor, Other doctors, Responsible Nurse, Other Nurses, Companion, Room Cleaner, Smart Patient Tracking Application

Subject Attributes: Responsible Doctor, Doctor, Nurse, Responsible Nurse, Companion, Room Cleaner, Smart Patient Tracking Application

Objects: Smart Door, Camera, Blood Pressure, EEG, EMG, Glucose Monitoring (CGM), Temperature Sensors.

Object attributes: Smart Door, Camera, Medical IoT Devices.

Environmental Attributes:

✓ Authentication: web/mobile app, personal card

✓ Urgency: urgent/non-urgent

✓ Time: cleaning hours, doctor visiting hours, nurse visiting hours, visiting hours

✓ Doctor Permission: approval/ deny

The scenario is detailed below:

Smart Door: Smart door can be opened with predefined authentication methods for the situations defined in my system. These are:

- The doctor in charge of patient observation is authorized to open the door in any case.
- Doctors other than the responsible doctor can only open the door during doctor's visiting hours using their hospital staff card.
- The door can be opened in any situation with the help of a button from inside.
- During the nurse patient visit, the nurse on duty can open the door via the personnel card and smart application.
- Other nurses can only open the door with their personal card in emergencies.
- The patient's companion can open the door in any situation with the companion card.
- During patient visiting hours, the smart door can be opened to everyone without any authentication.
- In emergencies, the door can be opened independently of the clock conditions with the smart patient tracking application.
- cleaning staff can only enter the room with their personal identity cards during cleaning hours.

Camera: With the smart health tracking application, the camera can only be watched by the doctor in charge and the nurse in charge only in emergency.

- Only in emergencies, the companion can monitor the camera with the smart tracking application.

Wearable Medical Devices: The smart health app is authorized to read device information.

4.4.3. Scenario 3 : Non-IoT Domain Daily Life Scenario :

The scenario in this section is designed to see how the CA-IRBAC algorithm can be used in non-IoT environments. The system is designed as an appointment tracking system where doctors who want to provide personalized online counseling services and those who want to get counseling from them can create and manage online appointments.

An appointment that can be controlled (created, canceled, etc.) according to different user types, such as objects in the IoT environment, and has different states is accepted as an object. Users can be people and applications like in IoT applications. In order to perform an operation related to the appointment, it is important who the user is, the time and the current status of the appointment. These take the place of attributes in IoT applications.

System Variables:

Subjects: Doctors, Patients, System administrator, Web application

Subject Attributes: Doctor, Patient, System administrator, Web application

Objects: Appointment

Object Attributes: Created, Cancelled, Completed, Not made

Environmental Attributes: Last Appointment Day: 3 days before or not - Payment Success: yes / no - Before the appointment: 2 hours or not - Before the appointment :12 hours or not - Appointment time: yes / no

Authentication: Web

Operations: Create, Cancel, Complete, Not made

4.4.3.1. General Structure:

- Doctors are defined by the authorized system administrator to the system.
- Doctors who will give consultancy to previously defined clinical units are registered by the system administrator.
- The appointment calendar, which includes the days and hours for each doctor to make an appointment, can be saved and updated in the system by the system administrator.

- According to the appointment calendar of the doctor from whom online consultation is requested, patients can request an appointment on the appropriate days and hours.
- Patients who want to make an appointment register in the system by filling out a form containing their personal information individually.

Client:

- A registered client can create an appointment for a doctor of his choice from the list of doctors in the clinics defined in the system.
- Only the days and hours at which they can make an appointment are listed for the client who will request an appointment.
- The client can see the appointment list that he has created before and can cancel the active appointments that are in Cancelable status.

Doctor:

A doctor registered in the system as a consultant can see his past and current active appointments by logging in.

4.4.3.2. Object Status:

Appointment Created:

The process of creating an appointment process starts with the user registering and logging in to the system and clicking the make appointment button.

- After selecting the relevant doctor, day and time, the person is directed to the payment page prepared for the payment of the online consultation fee.
- After the successful completion of the payment process, an appointment is created.
- A client cannot make more than one appointment on the same day. At least 3 days must have passed between two appointments.

Status: Appointment is created.

Appointment Cancelled:

- Appointment cancellations can happen in two ways.
- The doctor has the right to cancel the appointment up to 2 hours before.

- The counselee, on the other hand, has the right to cancel the appointment he has made up to 12 hours before the appointment time.
- Appointments with less than 12 hours remaining cannot be canceled by the client.
- A refund will be made for an appointment canceled by the consultant or the client.
- A refund process is initiated for a canceled appointment.

Status: Canceled

Appointment Completed:

- An appointment that has been created is in a created state unless it is canceled by the appointment time.
- In order for the appointment to be completed, the appointment time must have come for an appointment that has been created and is active.
- When the appointment time comes, the appointment is considered completed.
- It is the client's own responsibility to attend the online counseling meeting at the appointment time, and in the online meeting where the client is not attended, the appointment is considered completed and no refund will be made.

Status: Completed

Appointment Not-Made:

- If the online meeting could not take place or was interrupted due to a technical failure on the part of the consultant during the time of the online meeting, it can be updated to the status not realized by the consultant or the system administrator. Refund process is initiated for the missed appointment.

Status: Not made.

General Structure of Appointment Smart Contract:

```
contract Appointment{
      enum Status {Created, Cancelled, Completed, Not Made}
      struct Appointment {
        string id;
        Status status;
        string doctorID;
        string consultantID;
        uint256 createdDate;
        uint256 appointmentDate;
        uint256 cancelledDate;
        uint256 notMadeDate;
      }
      event AppointmentCancelled(string id, string cancellerID, uint256 date );
      event AppointmentCompleted(string id, uint256 date);
      event AppointmentNotMade(string id, uint256 date);
      address owner;
      mapping(string => Appointment) appointments;
      function add(string memory id, uint256 appointmentDate, string doctorID; string
    consultantID;)
            public onlyOwner {
                  orders[id] = Appointment(id, Status.Created, doctorID; consultantID;
                  getTime(), appointmentDate, 0, 0 );
            }
    }
    ( Other functions: get(), cancel(), complete(), notMade() )
```

4.4.4. Implementation in General

For this project, subjects, objects, operations and all their attributes are defined on MySQL Database. The web user interface is created to simulate operation that are accessible for different user levels.

During the project development process, it is assumed that smart contracts for each object can be created and updated by the super authority using a web interface.

4.4.5. Simulation Setup

The simulation demonstration is shown on the smart home scenario.

Web User Interface: To simulate process a web user interface developed on Apache web server then the mobile application and the biometric authentication options are simulated.

The defined authorized user types are listed after authentication choice on the screen. After logging into the system as an authenticated parent or babysitter, the authorized user reaches the list of the smart home devices that can be intervened. In the next stage, the user activates the operation request by selecting the action can be done for the selected device.

After the operation transaction request, the defined context along with the parameters required for the relevant operation transaction is pulled from the database and sent to the evaluation function defined in the smart contract via the web service to check the accessibility of the requested operation accessibility

The Steps are shown below:

The system implemented by assuming that the parent type user enters the system via the mobile application is simulated.

In order to use the smart home system, it is necessary to be authenticated in the system first. To simulate authentication, the web page is created and mobile authentication and biometric authentication options are offered to user.

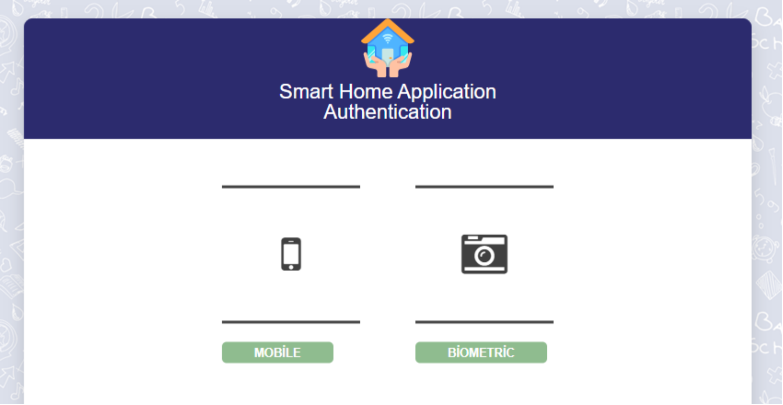
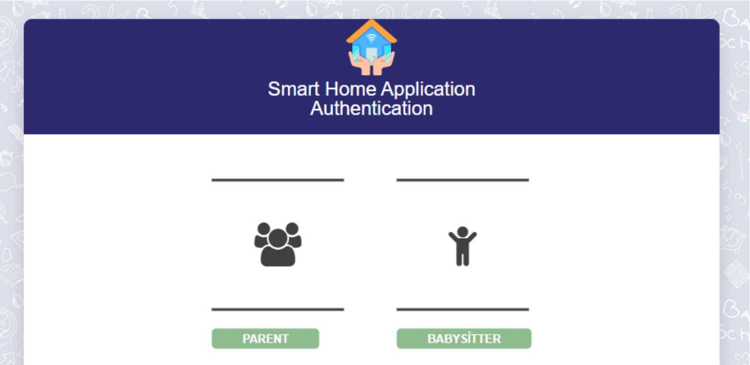


Figure 4.4.5.1. Authentication options are listed on the web screen.

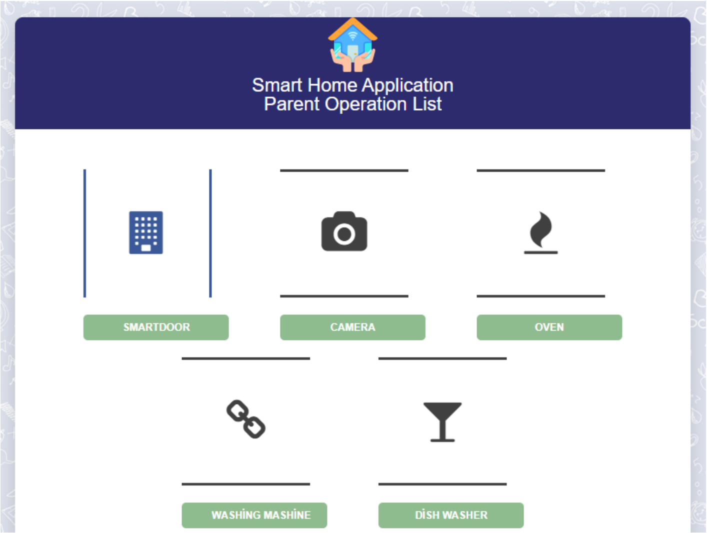
After the user selects the authentication method, the user types that can log in with the relevant authentication method are listed on the screen. In this simulation setup mobile authentication method was selected and relevant user types with this authentication method were listed as parent and babysitter.



Smart Home Application
Authentication

Figure 4.4.5.2. Mobile application users are listed.

It was assumed that the data required by the authentication methods related to the screen selections of the individuals, such as the username, password, and biometric data, are automatically verified.



Smart Home Application
Parent Operation List

Figure 4.4.5.3. The accessible parent operations are listed.

✓ In this scenario, the system assumed that the user was the parent and thought that it would open and close the smart door.

✓ After the selection of the desired operation, the request requirements are checked from the database.

✓ Context parameters of the targeted process are read from the database and sent to smart contracts for evaluation.

Pre-controls for the desired operation, person attributes, object attributes, such as information related to authentication type, are taken from the database and sent to the smart contract, evaluated according to the context rules, and the result value for the access permission is returned.

CHAPTER 5

RESULT AND CONCLUSION

As an implementation result of the access control algorithms used, the necessary permissions and context information for the objects are as given in the proceeding tables. When the systems are examined in terms of the created roles, permissions and contexts, the total security policy can be calculated with the formula given below.

With the formula used when calculating the access control system complexity, it was tried to obtain the number of security policies that were defined in the whole system and needed to be controlled. The large number of security policies will make the structure more complex.

The formula expresses all of the permissions given by the authentication method required for each context.

Formula Structure:

✓ Permissions defines the operation permissions that can be performed for each object.

Permissions = (Number of Objects) * (Number of Operations)

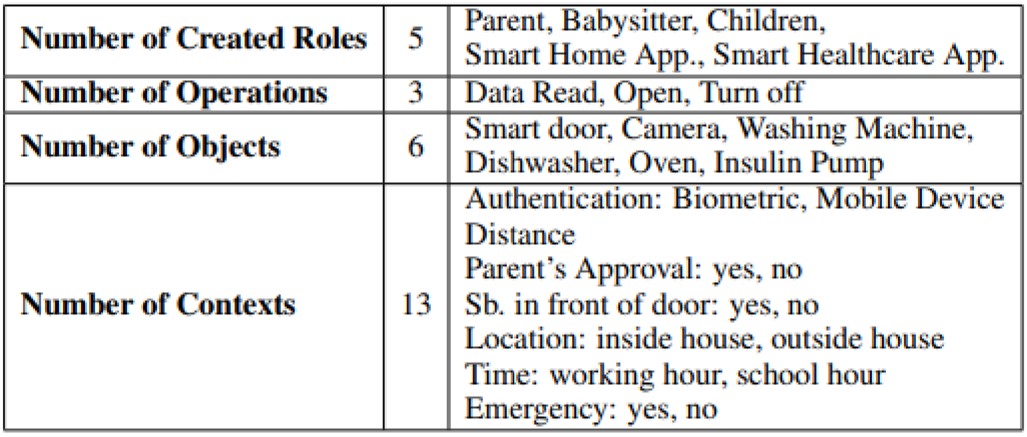
✓ Security Policies represents the sum of the security policies that will be operated on the basis of valid contexts with each valid authentication method of each role.

Security Policies = (Number of Roles) * (Number of Permissions) * (Number of Context)

5.1. RBAC Complexity

Permissions = (6 objects) ∗ (3 operations) = 18

Security Policies = (5 roles) ∗ (18 permissions) ∗ (13 context) = 1170

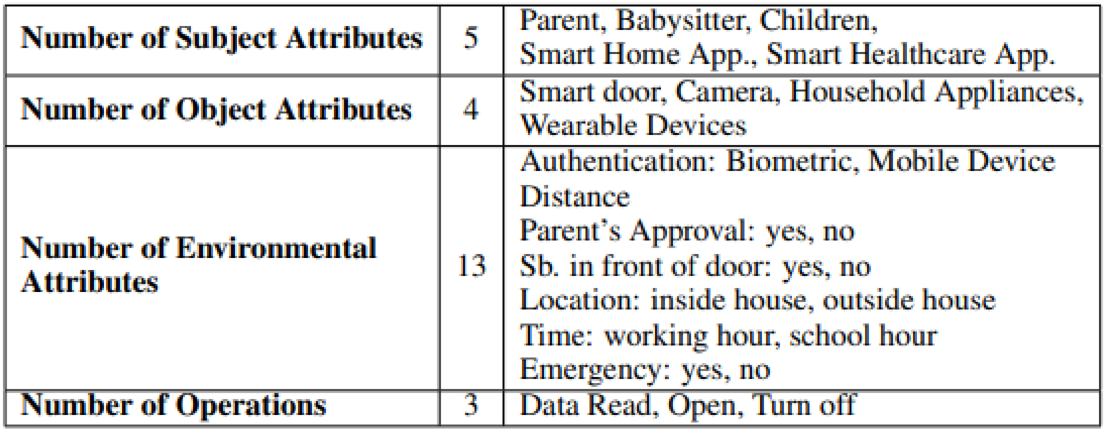


Number of Created Roles	5	Parent, Babysitter, Children, Smart Home App., Smart Healthcare App.
Number of Operations	3	Data Read, Open, Turn off
Number of Objects	6	Smart door, Camera, Washing Machine, Dishwasher, Oven, Insulin Pump
Number of Contexts	13	Authentication: Biometric, Mobile Device Distance Parent's Approval: yes, no Sb. in front of door: yes, no Location: inside house, outside house Time: working hour, school hour Emergency: yes, no

Figure 5.1.1. Complexity Analysis of CA-RBAC

5.2. ABAC Complexity

Security Policies = (4 OA) ∗ (5 SA) ∗ (13 EA) ∗ (3 OP) = 780



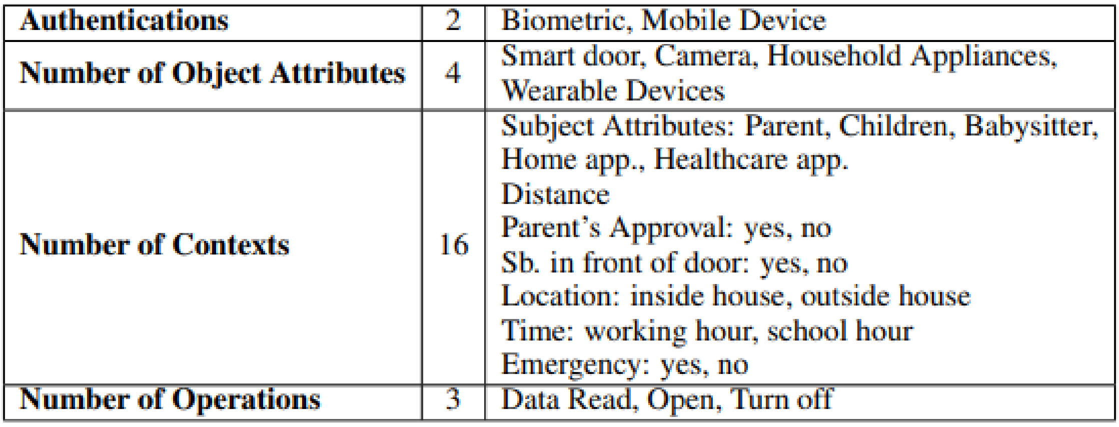
Number of Subject Attributes	5	Parent, Babysitter, Children, Smart Home App., Smart Healthcare App.
Number of Object Attributes	4	Smart door, Camera, Household Appliances, Wearable Devices
Number of Environmental Attributes	13	Authentication: Biometric, Mobile Device Distance Parent's Approval: yes, no Sb. in front of door: yes, no Location: inside house, outside house Time: working hour, school hour Emergency: yes, no
Number of Operations	3	Data Read, Open, Turn off

Figure 5.2.1. Complexity Analysis of ABAC

5.3. CA-IRBAC Complexity : Smart Home Scenario

Number of Context = (11 EA) + (5 SA) = 16

Security Policies = (3 Op) ∗ (2 Auth) ∗ (4 OA) ∗ (16 Context) = 384



Authentications	2	Biometric, Mobile Device
Number of Object Attributes	4	Smart door, Camera, Household Appliances, Wearable Devices
Number of Contexts	16	Subject Attributes: Parent, Children, Babysitter, Home app., Healthcare app. Distance Parent's Approval: yes, no Sb. in front of door: yes, no Location: inside house, outside house Time: working hour, school hour Emergency: yes, no
Number of Operations	3	Data Read, Open, Turn off

Figure 5.3.1. Complexity Analysis of CA-IRBAC

5.4. CA-IRBAC Complexity: Smart Patient Room Monitoring Scenario

Number of Context = (8 EA) + (7 SA) = 15

Security Policies = (2 Op) ∗ (2 Auth) ∗ (3 OA) ∗ (15 Context) = 180

Authentications	2	Web App / Personal Card
Number of Object Attributes	3	Smart Door, Camera, Medical IoT Devices
Number of Contexts	15	**Subject Attributes:** Doctor, Responsible Doctor, Nurse, Responsible Nurse, Companion, Room Cleaner, Smart Patient Monitoring Application **Emergency:** yes, no **Time:** cleaning hours, doctor visiting hours, nurse visiting hours, visiting hours **Doctor Approval:** yes, no
Number of Operations	2	Data Read, Open

Figure 5.4.1. Complexity Analysis of CA-IRBAC

5.5. CA-IRBAC Complexity : Non-IoT Scenario

Number of Context = (10 EA) + (4 SA) = 14

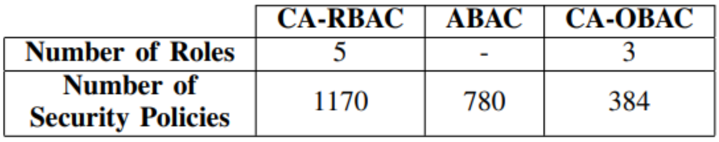
Security Policies = (4 Op) ∗ (1 Auth) ∗ (4 OA) ∗ (14 Context) = 224

Authentications	2	Web App
Number of Object Attributes	3	Created, Cancelled, Completed, Not made
Number of Contexts	14	**Subject Attributes**: Doctor, Patient, System Administrator, Web application **Last Appointment Day:** 3 days before or not **Payment Success**: yes ,no **Before the appointment:** 2 hours or not **Before the appointment :**12 hours or not **Appointment time:** yes , n
Number of Operations	4	Create, Cancel, Complete, Not made

Figure 5.5.1. Complexity Analysis of CA-IRBAC Non-IoT Scenario

5.6. Comparison of Access Control Algorithms

Figure 5.1, 5.2 and with 5.3 we can see the complexity of the security policies obtained by applying each access control algorithm. With the proposed method, it is seen that the complexity is reduced by 3 times. In the role-based algorithm, which has a role expansion problem, it is seen that this problem disappears with the use of operations instead of roles.



	CA-RBAC	ABAC	CA-OBAC
Number of Roles	5	-	3
Number of Security Policies	1170	780	384

Figure 5.6.1. Complexity of Algorithms

5.7. Smart Contract Contribution

According to the security policy created for each object in the project, the decision mechanism of evaluation transactions is on the Smart Contract. In addition to the implemented access control algorithm referred from [11] to in this study, the security policies of each object are controlled by Smart Contracts. Smart Contracts of objects are assumed to be deployable to the Ethereum private network by the authorized user. Object Smart Contracts are stored in an encrypted and secure shared ledger. The security gains can be listed as follows.

Privacy: It was tried to ensure that policy changes can only be made by the authorized person, and the security of access to IoT devices was ensured by preventing outside interventions.

Immutability: During the context evaluation, all necessary access parameters and the evaluation result are stored on the smart contract, and an immutable log information is automatically kept for access requests.

Single Point of Failure: Our model uses a distributed access control points, which eliminates the single point of failure.

5.8. Result and Conclusion

The main purpose of this thesis is to provide reliable and secure IoT environment with detailed and dynamic access control. The Context-Aware Role-Based Access Control (CA-IRBAC) model, which was developed by adding attributes to the Role-Based Access Control (RBAC) model, which was presented as an access control model suitable for the IoT environment with many-to-many interaction for the purpose, was implemented to work with a smart contract. In this operation-based model, operations are grouped instead of roles. First, it is checked whether the incoming requests match the requested operation. Roles have been reduced, so the number of max policies has decreased significantly. At the same time, since the policies were determined according to the attributes of the objects, it was not necessary to determine the policy for each object, which again led to a decrease in the number of security policies. Access control is dynamic using attributes. During the implementation of The CA-IRBAC algorithm, the security policy evaluation with smart contracts caused an extra processing power and cost, but it also contributed to a reliable IoT environment by ensuring that the policies cannot be changed without authorization. Thanks to the fact that Contracts are stored in a distributed way, not in a single center, a security problem that may occur at a single point does not prevent the operation of the system. The system and data have become resistant to technical failures and malicious attacks. Access control is provided in a reliable environment with Blockchain.

REFERENCES

1. "What is internet of things (IoT)? " Last Updated March, 2022, https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT

2. "Internet of Things statistics for 2022 - Taking Things Apart" Last Updated March 08,2022, https://dataprot.net/statistics/iot-statistics/

3. Crosby, M., Pattanayak, P., Verma, S. and Kalyanaraman, *V. Blockchain Technology: Beyond Bitcoin. Applied Innovation,* 2016, 2, 71.

4. Hassija, Vikas, Vinay Chamola, Vikas Saxena, Divyansh Jain, Pranav Goyal, ve Biplab Sikdar. 2019. "A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures". IEEE Access. Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/access.2019.2924045.

5. Hurlburt, George F., ve Irena Bojanova. 2014. "Bitcoin: Benefit or Curse?" IT Professional. Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/mitp.2014.28.

6. Al-Rakhami, Mabrook S., ve Majed Al-Mashari. 2020. "Blockchain and Internet of Things for Business Process Management: Theory, Challenges, and Key Success Factors". International Journal of Advanced Computer Science and Applications. The Science and Information Organization. https://doi.org/10.14569/ijacsa.2020.0111069.

7. Internet of Things (IoT) connected devices worldwide in 2019 and 2030, last updated Aug 6, 2021, https://www.statista.com/statistics/1183463/iot-connected-devices-worldwide-by-technology/

8. Pilkington, Marc. t.y. "Blockchain Technology: Principles and Applications". Research Handbook on Digital Transformations. Edward Elgar Publishing. https://doi.org/10.4337/9781784717766.00019.

9. Park, Ji-Sun, Taek-Young Youn, Hye-Bin Kim, Kyung-Hyune Rhee, ve Sang-Uk Shin. 2018. “Smart Contract-Based Review System for an IoT Data Marketplace”. Sensors. MDPI AG. https://doi.org/10.3390/s18103577.

10. R. Beck, J. S. Czepluch, N. Lollike, and S. Malone, “Blockchain-the gateway to trust-free cryptographic transactions,” in Proc. 24th Eur. Conf. Inf. Syst. (ECIS), Istanbul, Turkey, 2016, pp. 1–15.

11. Computer Security Division, Information Technology Laboratory (2016-05-24). "Attribute Based Access Control | CSRC | CSRC". CSRC | NIST. Retrieved 2021-11-25.

12. GENÇ, Didem. Context aware role based access control model for internet of things applications. 2018. PhD Thesis. Izmir Institute of Technology (Turkey).