A Blockchain and Capability Based Access Control for Internet of Things using Self-Sovereign Identity

by

Md. Safinur Amin
24341119
Abu Bakar Siddique Khan
20301041
Istihad Ahmed
22241144
Tawsif Mahamud Sajid
24341099
Md. Mohtasim Monir
24241187

A Thesis submitted to the Department of Computer Science and Engineering in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
December 2024

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Md. Safinur Amin
24341119

Abu Bakar Siddique Khan
20301041

Istihad Ahmed
22241144

Tawsif Mahamud Sajid
24341099

Md. Mohtasim Monir
24241187

Approval

The thesis titled "A Blockchain and Capability Based Access Control for Internet of Things using Self-Sovereign Identity" submitted by

1. Md. Safinur Amin (24341119)
2. Abu Bakar Siddique Khan (20301041)
3. Istihad Ahmed (22241144)
4. Tawsif Mahamud Sajid (24341099)
5. Md. Mohtasim Monir (24241187)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on November 30th, 2024.

Examining Committee:

Supervisor:
(Member)

Dr. Md Sadek Ferdous

Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi

Associate Professor and Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Access control systems are essential tools for businesses because they guarantee the safe management of user access to resources after verification. Conventional approaches, such as Role-Based Access Control (RBAC) and Discretionary Access Control (DAC), have drawbacks such complexity brought on by "role explosion" and susceptibility to Trojan horse assaults. Moreover Attribute-Based Access Control (ABAC) has scalability challenges. As the number of attributes increases, managing and maintaining the policies associated with these attributes can become complex. Another significant drawback is, ABAC relies heavily on policies that define the rules for granting or denying access based on attributes, introducing the challenge of policy management complexity. While it provides an alternative, Attribute-Based Access Control (ABAC) has drawbacks such as "attribute explosion" when the number of attributes rises. The lightweight and dynamic properties of Internet of Things (IoT) devices provide security challenges for centralized access control systems. To ensure that only authorized entities may connect with certain IoT resources, access control becomes essential to handle issues related to single-point failure, user authentication, and privacy leaks. To mitigate the risks of centralized approaches, a technique called distributed access control is suggested. It is investigated whether integrating blockchain technology might improve security. Benefits of blockchain include its decentralized, transparent, and unchangeable nature. Nevertheless, there are drawbacks to the current blockchain-based IoT access control solutions, such as their vulnerability to Distributed Denial of Service (DDoS) assaults. To solve these problems, the suggested method integrates capability-based access management with Self-Sovereign Identity (SSI) inside a blockchain environment. By ensuring that entities only have permissions that are in line with their roles and responsibilities, this method seeks to securely manage and transfer capabilities. The creation of a prototype system highlights the feasibility and effectiveness of the suggested solution in the research. This system provides a possible remedy for the security issues raised by IoT settings and blockchain technology by demonstrating the combination of SSI and capability-based access control in a practical setting.

Keywords: Self-Sovereign Identity (SSI); Blockchain; Capability Based Access Control (CapBAC); Internet of Things (IoT) devices

We extend our utmost gratitude to the Almighty Allah, without whom our thesis could not have been successfully completed.

Acknowledgement

Dr. Sadek Ferdous sir, Professor of Computer Science and Engineering Department and our supervisor, is thanked for his invaluable assistance and counsel regarding our project. He assisted us in times of need.

And lastly, our parents, without their unwavering support this endeavour might not have been feasible. Their thoughtful prayers and support have brought us to our commencement.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Access control has high importance in the ongoing development of the IoT due to its reputable security and adaptability. When introduced with the dynamic and heterogeneous attributes of IoT devices, normal systems collide with obstacles. Additionally, these systems also generate concerns about the privacy and vulnerable attributes to particular points of failure. To supervise the accessibility of IoT functionalities, an innovative method is suggested in our proposal.Conventional access control mechanisms frequently encounter difficulties in adjusting to the ever-changing and diverse characteristics of Internet of Things (IoT) environments. CBAC facilitates precise management of access permissions, permitting devices to possess and exhibit the necessary capabilities to retrieve particular resources [9].

Moreover Significant security concerns exist in addition to operational deficiencies that legacy access control systems are unable to adequately address. The implementation of the principle of least privilege appears to present a significant obstacle. Similar to right delegation with a documented audit trail and real-time, albeit transient, right issuance and revocation, the confused deputy problem remains an unresolved matter.The integrity of access control decisions can be improved by leveraging the immutability and transparency of the blockchain, thereby reducing the likelihood of malevolent activities and unauthorized access. This malleable strategy is consistent with the dynamic nature of IoT ecosystems.A multitude of security hazards can compromise IoT devices, such as unauthorized access, data manipulation, and device impersonation. Prevalent centralized access control models are insufficient for the decentralized and dispersed characteristics of IoT networks due to their susceptibility to single points of failure [8].

The implementation of SSI represents a significant departure from the status quo in identity management, as it empowers users to manage their digital identities independently of a central authority. The incorporation of SSI into IoT access control guarantees users autonomy over their identities, thereby bolstering confidence and confidentiality within the ecosystem [7].Through the integration of Blockchain, CBAC, and SSI, this study endeavors to develop an all-encompassing and holistic strategy to tackle the complex security issues that arise in IoT environments. The integration of these technologies aims to establish an ecosystem that facilitates

secure and efficient interaction among devices, users, and data [14].This method adopts the use of blockchain technology and self-sovereign identification (SSI).

1.2 Research Objective

This technique employs capability tokens to govern data access by smart contracts that are implemented on the Ethereum platform. Giving the permission for the modification of user and organisation identities and attributes through SSI implementation leads to the improvement in privacy and autonomy. The collective graph keeps authority over token production, task delegation, execution, and cancellation, while smart contracts safeguard alliances to ensure impartiality and efficient way of verification.

- R.O:1. We need to ensure that we do have in-depth knowledge about the existing researches about blockchain and access control and identify the research gaps.
- R.O:2. After identification we are proposing a system by which we can mitigate the gaps.
- R.O:3. A capability token care method based on smart contracts making sure of tokens being safe, functional, and long-lasting.
- R.O:4.Safeguarding tokens against abuse and unauthorized use.
- R.O:5. An process of token testing that uses delegation graphs to resolve problems that arises from inconsistent or conflicting capability tokens. It simplifies the authentication process.
- R.O:6. A deep evaluation of the viability, scalability, safety, and functionality of the mentioned approach in comparison to the methods that already exist.
- R.O:7.Enable decentralized, fluid, and adaptable access control for both users and devices.

1.3 Report Structure

In chapter 2, we present the background of our research. Here we talk about the brief history and give brief knowledge about the techniques and components (e.g. Blockchain, Access Control, IoT, SSI) that we used for our research. Subsequently, in chapter 3, we gave literature reviews of the papers that we studied to gain knowledge for our research. Later on, in chapter 4, we provided the methodology of our research. This chapter provides an overview of our work. Finally, with chapter 5, we concluded our report by giving a brief about the whole research.

Chapter 2

Background

2.1 Access Control

A process that enforces security policy by the proper utilization of system resources with authorization from the authorized entity, access control is the core of the security system for information technology (IT) devices. Safeguarding data and resources from illicit access (secrecy) and inappropriate or alternation (integrity) while ensuring their availability to authorized users (no denials-of-service) is a critical requirement for any information management system. The administration system of a security policy by Access Control establishes the authorized entities (including processes and individuals) and the degree of access they are granted to the system resources. It is a critical security policy that causes reduction of the likelihood of harm to the organization or business. Access controls function by distinguishing an individual or entity, validating the identity or nature of the claimed user or application, and granting authorization for the degree of access and specific operations that are linked to the username or IP address.

2.1.1 Mandatory Access Control

In this particular domain of Access Control Management, the capacity of an individual to authorize or refuse access is limited. The operating system adheres rigorously to the criteria that are established by the system administrator so that the system can function properly.

A centralized framework of parameters and policies is preserved by MAC. Each potential entrant is assigned to a distinct category under this system, which enables the denial of access [1]. The authority for establishing and modifying access policies and permissions for various areas of the premises is vested in a centralized security administration, over which the end-users have no say and are unable to make alterations. The users are not allowed to alter the access rules and permissions, which are rigorously enforced by the system administrators and are defined manually. High-risk properties housing sensitive data and information, such as government buildings, healthcare facilities, banking and financial institutions, and military initiatives, are the optimal locations for these systems.

By restricting modification of the pre-established criteria to the centralized system

administrator, the system ensures optimal security. Nevertheless, in the event of user changes, routine updates are necessary and are executed manually by the system administrator. Additionally, it is inflexible and time intensive. The implementation can be challenging and intricate at times.

To ascertain the authorization decision d, a MAC rule r is a declarative statement that executes "if c then d," with system states (global system events) and AC characteristics (subjects, objects, or actions) serving as the constraint c [1]. "If (a user is a part of the group X with a security level of 3, the current day is Friday, the user's action is to read, and the object is file Y), then grant," is a rule that serves as an illustration [1].

The tuple (s, o, a) denotes an access request q, in which $s \subset S$, $o \subset O$ and $a \subset A$. A query (s, o, a) signifies that the subject s wishes for the object o to perform the specified action. Each of the variables S, A, and O is capable of possessing multiple attributes.

The MAC model comprises a sequence of regulations, each of which is mathematically represented by the following c expression:

$$(sCond, oCond, aCond, decision, gCond, st).$$

The constraints $sCond$, $aCond$, and $oCond$, respectively, apply to the attributes of a subject, action, and object. $gCond$ is an all-encompassing constraint that can include any attribute of objects, subjects, actions, or systems (e.g., the present time and system load). When all conditions st, $oCond$, $aCond$, $gCond$, and s are assessed as TRUE for a given request (s, o, a), the request is deemed authorized or denied in accordance with the decision d.

Therefore, the applicability of each rule to a given request is expressed as follows:

If

$$(sCond \wedge oCond \wedge aCond \wedge gCond \wedge st) \rightarrow d$$

In a rule, more complex constraint structures can be specified. For instance, principles that are applicable to queries containing multiple attributes can be specified.

A finite state transducer that is deterministic and corresponds to a MAC model:

Finite state Machine with 5 tuples:

$$M = (\Sigma, ST, s_0, \delta, F)$$

where:

$$\begin{aligned}\Sigma = \{&sCond_1, \ldots, sCond_n,\\ &oCond_1, \ldots, oCond_n,\\ &aCond_1, \ldots, aCond_n,\\ &gCond_1, \ldots, gCond_n\}\end{aligned}$$

The input alphabet consists of the attribute constraints that are linked to the following components:

access (a)
global event (g)
object (o)
subject (s)

$$S = \{st_0, st_1, st_2, \ldots, st_n, \text{Grant}, \text{Deny}\}$$

consists of a recorded set of finite, non-empty MAC system states and permissions. st_0 denotes the starting condition.

The state-transition function is δ, where

$$ST \times \Sigma \to ST$$

The set of final states is $F = \{\text{Grant}, \text{Deny}\}$.

2.1.2 Discretionary Access Control

This is an access control method in which the policies defining who or what has been granted permission to access the resource are established by the proprietors or administrators of the protected system, data, or resource. The management of access permissions falls under the purview of the business owner within this domain of Access Control Management. In this instance, the operating system does not decide regarding the permissibility of denying entry based on any criteria. It is mandatory for business proprietors to possess a comprehensive understanding of security policies and optimal methodologies. With Discretionary Access Control (DAC), one could grant permissions to whoever one desires. The security level settings can be ascertained by the end-user through the delegation of access to others, which may involve lending their key card or disclosing their code. The user-friendliness and adaptability of this system are its greatest advantages, particularly considering the ease with which data can be maintained and access privileges granted. However, it is worth noting that users can delete files while simultaneously viewing and writing. This feature may render it less secure compared to a MAC, as it increases the risk of information disclosure. Additionally, it is not centralized, unlike MAC, which renders monitoring the data traffic and the authorization of privileges unfeasible.

Denote the sets S, O, and P as the collection of subjects, objects, and permissions, respectively. An enumeration of access control elements may be represented by the set

$$A \subseteq S \times O \times P$$

As permissions are revoked, triplets are expunged from A. Triplets are appended to A whenever new permissions are granted.

Access Control Representation

The columns of an access control matrix $M_{s,o}$ represent objects, while the rows represent subjects.

$$M_{s,o} \subseteq P$$

represents the collection of permissions granted to object o by subject s.

Access Control Lists

An access control list is a set $\{A_o \mid o \in O\}$, one element for each object. The elements of the list are the pairs (s, p) of subjects s who have permission p to that object.

Capabilities

In order to perform capability storage, identifiers are allocated to each subject, which subsequently confer access to the appropriate permissions.

```
int fd = open("/etc/passwd", O_RDWR);
```

$\Rightarrow$ fork() + exec(), new process inherits fd (the authorization "token")

2.1.3 Role Based Access Control

This is a prevalent access control mechanism that imposes limitations on computer resource access in accordance with the business functions of specific individuals or groups. The administrator can implement this approach to limit access to privileged information in accordance with an individual's job responsibilities, authority, and competency.

The system administrator does not have to provide access rights to different users inside the system; rather, they just assign permissions to the precise job responsibilities and titles of the employees. Even though the system provides flexibility, identifying roles may be difficult in big businesses that have a wide variety of job tasks, and assigning them can be a time-consuming process. This might result in a rise in the number of roles, which would in turn cause a "role explosion." When temporary permissions are granted to users, it is convenient to forget to remove such permissions at a later time. And it is a more effective approach for companies of a smaller scale.

The simplest RBAC model is a flat one. RBAC is an alternative designation for RBAC0, denoting the absence of a hierarchical structure among the roles.Hierarchical RBAC definition is formally as follows:

$$UA \subseteq U \times R, \quad PA \subseteq R \times P$$

RH could potentially be construed as a partial order on R, given the context. The classification of the relation R as a partial order relationship is predicated on the following attributes: asymmetry (roles lack the ability to exchange levels), transitivity (if a certain role is higher than a different one and that role sits above it, then the first role is also higher than the last), and flexibility (each role is either at the identical level as another or on a higher level than another).

Additionally, the symbol $\geq$ may be utilised to denote the role hierarchy. The proposition

$$r_1 > r_2$$

indicates that if both r_1 and r_2 are distinct roles within the hierarchy ($r_1, r_2 \in R$), then r_1 holds a superior position or is more advanced than r_2. When (u, r) is a member of the set of users (UA), a specific user u is regarded as an explicit member of the role r. Conversely, a user u is deemed to be an implicit member of the role r if (u, r') is a member of the set of users (UA) and there exists a set of users (R) in which $r' \geq r$. Permission p is granted to a user u if a role r in R ensures that (p, r) is in PA and u is an explicit or implicit member of r. The situation becomes such when UA and PA are furnished.

2.1.4 Attribute Based Access Control

Attribute-based access control is an approach to access control that processes authorization decisions regarding the granting or denial of permissions based on the characteristics ascribed to a subject, user, or object. In comparison to other access control systems, the Access Behavior Analysis (ABAC) mechanism is considerably more sensitive to the external environments in which it operates and is therefore more dynamic.

ABAC permission and access policies are determined by specific attributes and take context into account. This feature obviates the necessity for multiple roles and empowers system administrators to modify and update attributes without the need to rewrite the policy. As a consequence of requiring considerably fewer roles, ABAC simplifies the process of identity management. Despite having fewer administrative requirements, ABAC is quite complex to manage. The increased complexity of designing and implementing ABAC, which is resource-intensive, time-consuming, and less scalable, must be managed by administrators. Additionally, it is difficult to audit.

The making of the implemented ABAC model is described in detail. ABAC model governance is determined by access policies and attributes. The translation of the access policy into the access structure is required. As for the definition of access structure, determining whether the attributes provided by the requester satisfy the access requirements of the target is an efficient process.

The representation of the access structure is an access tree comprising leaf nodes and non-leaf nodes. In the tree, a threshold gate k and the number of offspring nodes n are denoted by each non-leaf node, with

$$0 \leq k \leq n$$

The node is regarded as an OR gate when k equals 1, and as an AND gate when k equals n. Every leaf node is comprised of an att_i value representing one of the attributes specified in the access policy and a threshold value $k = 1$ [11].

Through the implementation of a collaboration node within the access tree. The collaboration node (CN) functions as a terminal node, as intended. The establishment of controlled and verifiable collaboration is accomplished via a pivotal alteration wherein CN retains the identity of the group, represented as GroupId. GroupId is intended to restrict collaboration to members of a particular group. It is allowed for devices that are part of the same group to provide collaborative attributes in order to aid the requester in obtaining secure authorization [11].

Represent T as an access tree in which the root node is represented by γ. Following that, a successful algorithm, denoted as Satisfy(L), was developed with the purpose of ascertaining whether a provided set of attributes L adheres to the access tree T. The algorithm is executed recursively, starting from the root node γ [9]. In order to obtain the $\text{Satisfy}_x(L)$ value for a non-leaf node x, the $\text{Satisfy}'_x(L)$ value for each of the progeny x' of node x is computed. Satisfy is executed and returns TRUE if at least k_x of the progeny nodes returned by $\text{Satisfy}'_x(L)$ are TRUE [9]. The $\text{Satisfy}_x(L)$ function returns TRUE exclusively if att_i is a member of L when x is a leaf node. Additionally, with respect to the collaborating node x_C, the verification of the att_i provider's membership in the GroupId is facilitated through the utilisation of the GroupId stored within x_C [9]. Moreover, $\text{Satisfy}_{x_C}()$ yields the value TRUE in this circumstance.

2.1.5 Capability Based Access Control

Frequently abbreviated as a "key," capability is the sole element that authorizes access to a given resource and serves to designate it [8]. The capability serves as an immutable symbol of credibility. A capability is a key or talisman that represents a privilege that has been bestowed upon its possessor. In order to execute the action linked to the token, the requester must transmit both the request and the token to the gatekeeper/provider [8]. This entity is solely responsible for verifying the token's veracity. This typically entails verifying the integrity of the token's digital signature [8]. Should the token remain legitimate and undamaged, the request can be processed without the need for additional authentication of the requester or verification of its privileges [8].
These procedures have already been executed during the token's preparation and issuance by the authorization server. This process streamlines the system by eliminating the need to maintain a user list or access rules at the access control point, a responsibility that can be handled by a device with limited resources.

$$ICap_O[VID_S] = \{OP\}$$
$$IDC_o[VID_s] = \{VID_p, \{VID_{Ch}\}, Dep\}$$

In order to oversee the subjects' authorized actions for each object, the BlendCAC scheme establishes ICap and IDC tokens [6]. An IDC token

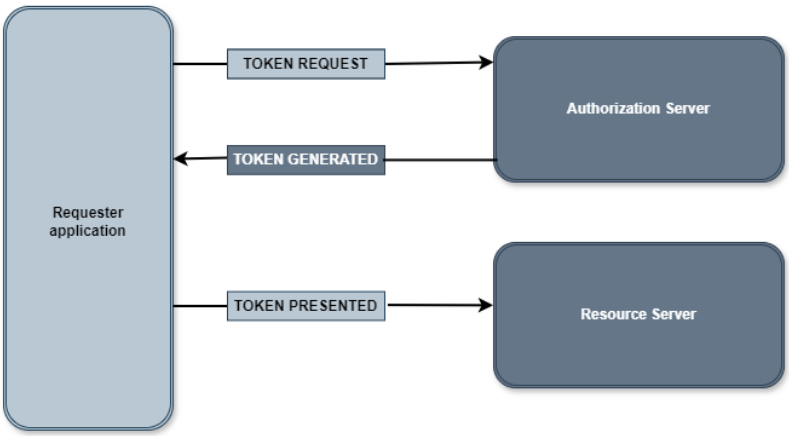

Figure 2.1: Issue of token in capability based access control (CBAC)

documents the delegation relationships between subjects for ordered actions, whereas an ICap token documents those actions themselves. The subsequent statements depict the data structures of an IDC token and an ICap token pertaining to a specific subject S [6].

2.2 Internet of Things (IoT)

The Internet of Things (IoT) comprises a network of devices that are interconnected capable of autonomously gathering, processing, and exchanging data, without the need for human intervention. Several industries, including transportation, medical care, agriculture, and smart communities, may utilise IoT devices to implement a variety of intelligent services. Access control is one area in which the Internet of Things does, nevertheless, present substantial security and privacy risks. Access control refers to the systematic process through which the characteristics and identities of the requester determine whether resources, such as devices, information, and services, are authorised or denied usage. Traditional approaches to access control, such as attribute-based access control (ABAC) and role-based access control (RBAC), rely on a centralised authority to prescribe access options and enforce access policies. The aforementioned methodology is considered unsuitable in light of the extensive, dynamic, and diverse characteristics of the Internet of Things (IoT), in which devices operate under varying trust presumptions, possess distinct functionalities, and are associated with multiple domains. Furthermore, centralised access management can harm the privacy of users and devices, hinder performance, and create a single point of failure [4].

2.2.1 Importance of Access Control in IoT devices

Access control in the Internet of Things is vital for guaranteeing the safety and privacy of sensitive information and resources that are shared and accessed by IoT gadgets and their users. This technology makes it easy for unique services and applications to be created that can use the data and capabilities of IoT devices across a variety of factories, including healthcare, industries, developed cities, and

residences. It can prevent misuse and unauthorized access of the IoT networks and devices safe, which can often lead to malicious activity, identity being stolen, or even data breaches. Furthermore, it also keeps private and sensitive data safe, which otherwise could be hacked or altered by IoT devices or other parties. Finally, this will encourage users and stakeholders to have more faith and belief in IoT systems and applications, which can drastically improve users' happiness and online experience. The uniqueness, scarce resources, mobility, dynamism, scalability, interoperability, and ethical aspects of IoT scenarios present many difficulties and require access control. In the end, these fosters raised Proper models, architectures and protocols are needed for access control in order to define and keep up the rights of access of devices and users in IoT. Given below are some of the examples of current IoT access control approaches and frameworks:

- RBAC is an access control system in which users and devices are given roles according to their respective functions and responsibilities.
- ABAC is an access management system that gives authorizations on the basis of the attributes possessed by users, devices, data and context.
- CBAC is an access control system that issues data credentials or certificates that contains the authorized access privileges of devices and users.
- PBAC is an access control method that manages access control decisions and rules through the use of policies.

Consequently, access control in IoT is a vital and tough undertaking that needs constant research and regular updates in order to accommodate the latest IoT technologies and scenarios.

2.3 Blockchain

A decentralised ledger technology, eliminates the need for a trusted intermediary and facilitates transactions between multiple entities in a secure and transparent manner, Blockchain technology offers several advantages in terms of data storage, including traceability, accountability, and immutability. On the other hand, smart contracts are self-governing programmes that have the ability to encode and enforce logic and regulations concerning transactions. Using blockchain technology, it is possible to implement decentralised IoT access control mechanisms in which confirmation of device-to-device sharing of access choices as well as regulations is performed via network consensus. One advantage of access control based on blockchain technology is its ability to facilitate self-sovereign identity, which allows entities and individuals to independently possess and manage their own identities and attributes without dependence on a centralised governing body. SSI implementation can improve the compatibility, privacy, and autonomy of IoT devices and consumers.

2.4 Self-Sovereign Identity (SSI)

The primary objective of SSI is to empower individuals with authority over their data and eradicate the necessity for intermediaries, including corporations and governments, to administer and retain this information [16]. Self-Sovereign Identity (SSI) provides users with control over their data through a decentralised model of identity management. SSI empowers individuals to establish and retain control over their digital identities, in contrast to conventional frameworks that entrust the storage and management of personal data to central authorities (e.g., governments or corporations) [16]. This digital identity is stored on a blockchain, a decentralised and secure ledger technology, to which we and authorised third parties have access with our permission.

The first goal of SSI is to protect everyone's data. By using self-generating identifiers, such as bio-metrics, usernames, and passwords, users can develop and utilize their own identity without being dependent on other external companies, such as, Google or Facebook, for data storage and verification [16].

Moreover, SSI lets users to safely share their data with various websites, services, and applications by using verifiable credentials that help verify the user's identity. SSI comprises decentralized technologies like blockchain to provide the system's safety, privacy, and interoperability [16].

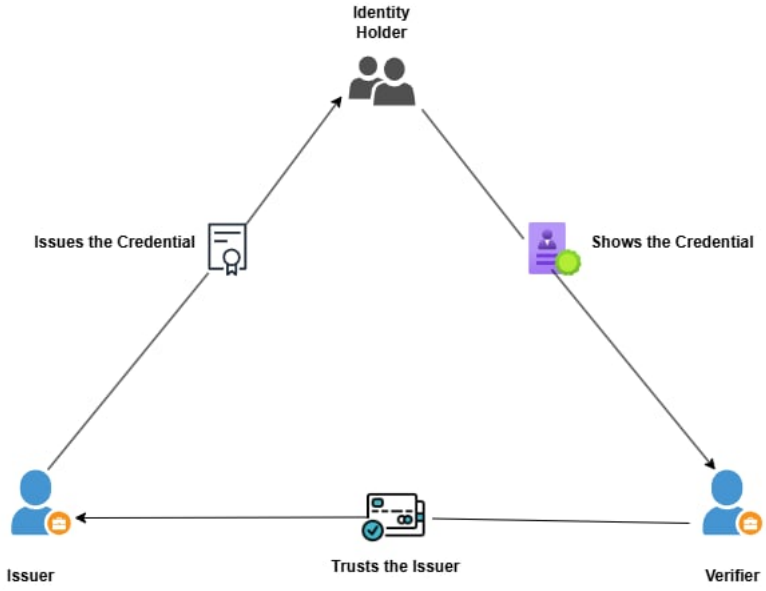

Figure 2.2: Identity Trust Triangle of Self-Sovereign Identity (SSI)

The Trust Triangle, when applied to the domain of self-sovereign identity, is a conceptual framework that extrapolates principles of social and commercial conduct from the physical to the digital realm. This is achieved through the implementation of a trust layer constructed atop the blockchain, in conjunction with protocols that serve as trustless intermediaries for operations typically carried out by governments, businesses, and individuals.

The triangle of trust, which defines a three-way relationship between parties (issuers, holders, and verifiers), serves as the foundational mechanism through which human trust is transmitted in business and social interactions. These three entities establish a dependable and secure digital transaction environment by operating in concert.

Originators of credentials are issuers, which include universities, government agencies, financial institutions, and corporations, among others.

Credentials are obtained by holders (customers) from issuers and stored in their digital accounts. When verifiers seek proofs of claims from one or more credentials, the credentials are presented. Subsequently, the verifier shall assess the issuer's digital signature for validity.
SSI has a many benefits, some of them are as follows:

- Gives user the choice of what data to disclose and with whom, and for what period of time. Which protects their privacy and anonymity.
- Reduces the possibility of fraud and identity theft by not saving user data on any centralized servers.
- Makes the use of several services and apps with a single identity, improving ease and user experience possible.
- It eliminates the necessity for multiple accounts and passwords.
- It encourages the user to be innovative and creative by enabling them to develop and utilize novel credentials and applications that exploit the capabilities and data of the identity system.

SSI encounters several challenges, among which decentralized key management stands out as a significant impediment to its market success.

- User education and cognizance are imperative, as individuals must comprehend the responsible and secure utilization and management of their identity and data.
- Building the new SSI ecosystem by establishing mutual acceptance of digital credentials among governments, industries, and other ecosystems.
- It necessitates the user's engagement and cooperation, as they are required to collaborate with other users and entities within the identity system and adhere to the established protocols and standards.
- It requires the user to oversee and comply with the legal and ethical regulations and standards that are applicable to their identity and data across various domains and contexts.
- Offline access for identity verification in the absence of an internet connection.

SSI is a nascent technological advancement that seeks to revolutionize the digital identity domain by granting users authority over their online personas.

Chapter 3

Literature Review

In the article, the authors talk about, a blockchain-based architecture is proposed as the solution, which addresses authentication and access control simultaneously [9]. Moreover, Blockchain-Enabled Capability-Based Access Control, which employs capabilities as credentials to grant secure and adaptable access rights, is among the most important characteristics. An attribute that sets blockchain apart is its ability to operate in conjunction with resource-constrained IoT devices in a manner that is both lightweight and efficient. The authors also mentioned, the primary objectives of the architecture are to satisfy critical characteristics of the Internet of Things, including usability, scalability, interoperability, and security. Finally, the paper ended by mentioning the operation of the system is maintained by separating the blockchain from tiny IoT devices and ensuring that they are lightweight and simple to use.

In this paper [8], the authors work with a prevalent access control mechanism that imposes limitations on computer resource access in accordance with the business functions of specific individuals or groups. They mentioned that, the administrator can restrict access to privileged information using this method. They further said that, subsequent access control decisions are regulated by capability metrics implemented during authentication, which facilitates the optimisation of task allocation between low-intensive operations on Internet of Things devices and intensive tasks on cloud servers.

They also provided a thorough assessment of security measures exposes the model's resilience to common attack vectors found in IoT environments, thereby guaranteeing the framework's security. They finally mentioned that using with the help of their work, organisations can have assurance that the information they obtain is precise, secure, and under the individual's control, given that the digital identity is entirely under the control and possession of the individual.

The study conducted in this paper [7] examines the application of verifiable credentials and decentralised identifiers within the SSI framework, with the objective of granting users increased authority over their personal information and reducing risks associated with their identities. This paper presents the Self-Sovereign Identity Based Access Control (SSIBAC) model, which integrates blockchain technology with conventional access control models to facilitate centralised authorization and decentralised authentication in cross-organization identity management [7].Finally,

the authors ensured that their proposed model places significant emphasis on data minimization during the access control process, thereby prioritising privacy.

This article [15] talks about it being an initial stride towards a world in which digital devices are secure and only the proper individuals may utilise them. It is a crucial research paper, because everyone desires their devices to be secure from malevolent individuals, and this paper helped pave the way for further knowledge about this particular topic. This relates to a novel approach to securing every intelligent device in existence. Smart devices. Electronic devices such as watches, refrigerators, and vehicles that establish an internet connection require safeguards against malicious individuals who may attempt to exert control over them. The authors implemented Ethereum, Blockchain technology, akin to an impenetrable magic book, enabling transactions to Create unique identifiers for these devices. The article also discusses the application of CapBAC to Ethereum. The authors made the intriguing decision to disassemble the keys into smaller components. Each component allows for a single operation with a device, such as turning on a lamp or inspecting the temperature of a refrigerator. Although the concept is intelligent, it is novel and potentially intricate. They also mentioned, it must be quick enough to manage multiple devices simultaneously.

In this article [14], the authors offer a unique design or model for IoT consortium networks. The authors also argue about the traditional methods of access control being vulnerable and weak to data leakage and single point failure. The authors mention their motivation, objectives and the architecture of their proposed model clearly. They provided a thorough and rigorous analysis and evaluation of their model. They also successfully showed its superiority over the already existing models. However, in conclusion, there are still some room for improvements as they left a few questions unanswered in their paper. The paper can benefit from more practical and user-oriented experiments.

The authors of this paper [10] gave the proposal for a novel architecture that combines Blockchain and SSI. Here, they used SSI for the security and privacy of their dsta emissions from smart vehicles. The authors talk about their aims and objectives of this research being decentralization and eliminating single point failure. Furthermore, they added about the users having control over their own identities and some other minor reasons. The authors go on to provide descriptions about processes needed to integrate Hyperledger Indy SSIs to detect and authorize users on Hyperledger Fabric network. The paper is very well written and structured, with clear objectives, motivations, background knowledge and results of the experiments. However, the authors did not provide any performance analysis of their proposed model in this paper which it could benefit from. Future research will definitely benefit this model described in this paper.

This paper [5] presents a new design approach for Access Control services leveraging smart contracts provided by blockchain technology. The main advantage of the authors' proposal is that the policy evaluation process is performed by smart contracts on a blockchain. This allowed the decision system to inherit the technological advantages of Blockchain. Usually, although any user can prove the access right to be granted by the resource owner when done fraudulently, in their reference

implementation, the resource owner could still fraudulently ignore access requests. Another disadvantage that was pointed out by the authors is that blockchain is still never used by a lot of users. The lack of user familiarity may be an issue for initial widespread adoption. Many things were well explained in the paper, however, they failed to address a good amount of issues (e.g. scalability issues is one of them). Even though the idea proposed by the authors is great, they failed to address many issues.

In this paper [5], we are first introduced to capability based access control. The authors talk about the previous approaches, and how they fail to provide easier privilege access. The authors applied CBAC on IoT devices. The limitation they discovered was that the object resource must be uniquely recognised and capable of being acted upon. An inherent limitation of capability-based permission is the necessity to distribute capabilities to every subject. There are also some drawbacks to this paper. The authors didn't focus on the complexities and scalability of the CBAC in IoT devices. IoT devices often have limited processing power, memory, and storage. Implementing CBAC can impact the performance and efficiency, which was not talked about in the paper. In conclusion, the paper made valuable contribution to its field, but still has room for improvements.

The paper [2] presents and examines an innovative method to augment the security and confidentiality of IoT devices and networks. The authors provide a combination of DIDs (self-sovereign and verifiable identifiers) and PoM (cryptographic protocol). The authors provide an argument that, this integration has the potential to establish confidence among Internet of Things (IoT) devices and networks, while also safeguarding them from malevolent attacks and unauthorised entry. The authors conclude the paper by mentioning that their proposed method is not only feasible but also very effective. The paper is well written and structured that provides a clear description on the problem, current level of knowledge, and the proposed model. The paper also provides a detailed description on the technical aspects. The study additionally assesses the approach from many viewpoints, including security, privacy, performance, and flexibility, and contrasts it with preexisting alternatives. The authors also discussed about the limitations and challenges to this approach adding some suggestions for future works. The paper is a vital contribution to the field of IoT security and privacy. It provides a solid theoretical and empirical foundation. However, the paper could benefit from more details on DID models and PoM protocols in practice including some diagrams tables and figures. They also could have also provided more evidence on the security and privacy properties on their approach.

In Table 3.1 and 3.2, we present a comparative investigation of relevant research. This table highlights various properties and features analyzed in different papers. In Table 3.2, we continue the comparative analysis and include the additional research papers.

Comparative Analysis (Part 1)						
Paper and Properties	Blockchain	SSI	Capability Based Access	Requirement Analysis	Threat Model	Literature Review
Sivaselvan et al. [9]	✓	✗	✓	✓	✓	✓
Bhat et al. [8]	✓	✓	✓	✓	✗	✓
Belchior et al. [7]	✓	✓	✓	✓	✗	✓
Amine et al. [14]	✓	✓	✓	✓	✗	✓
Rao et al. [15]	✓	✓	✓	✗	✗	✓

Table 3.1: Comparative investigation of relevant research (Part 1). A tick mark (✓) indicates the presence of a property, while a cross mark (✗) indicates its absence.

Comparative Analysis (Part 2)						
Paper and Properties	Blockchain	SSI	Capability Based Access	Requirement Analysis	Threat Model	Literature Review
Terzi et al. [10]	✓	✓	✗	✓	✓	✗
Maesa et al. [5]	✓	✗	✓	✗	✓	✓
Proposed system	✓	✓	✓	✓	✓	✓

Table 3.2: Comparative investigation of relevant research (Part 2). A tick mark (✓) indicates the presence of a property, while a cross mark (✗) indicates its absence.

Chapter 4

Proposal

4.1 Proposal

In order to establish themselves as the owners of the security cameras, householders produce ownership credentials for the cameras by utilizing a decentralized identifier registry (DID) [13]. These login credentials are tokens that allow certain access rights to the camera services. However, tenants are required to register their DID and get capability credentials. Once the authenticity of the request has been established, the owner of the camera will next supply the tenant with a capability credential that contains information regarding the authorized access privileges [13]. According to the principles of SSI, information might be disclosed in a selected manner. For the purpose of ensuring privacy, the renter just needs to supply the essential information when seeking access to the camera system. In turn, the owner of the camera receives information that may be verified without exposure to unnecessary risk. In the event that there is a threat to the security of the building or if the tenant no longer requires access, the owner of the camera has the power to immediately withdraw the capability credentials [13]. Because of this, there is no longer any chance of unauthorized entry or continuous tracking.
The purpose of this research is to propose the incorporation of a Self-Sovereign Identity and Capability-Based Access Model into smart home security systems. Integrating this way is in line with current security paradigms that value decentralization and user-friendliness. To hasten the creation of safe and privacy-conscious smart home technologies, this initiative aims to create an access management environment that is both more secure and more user-controlled. Making the environment safer will allow us to accomplish this.

4.2 Steps

For developing our proposed system, we need to follow up some steps.There are five steps which are threat modelling, requirement analysis, architecture design, protocol flow & implementation.

4.2.1 Threat Modelling

We have used the widely-known STRIDE danger model to draw attention to the risks associated with our investigation. Several separate security issues are considered, as

detailed in the list provided under this category [3].

- **T1. Spoofing Identity:** The act of unauthorized individuals attempting to enter a property by posing as a homeowner or renter. This approach will mitigate the threat of T1.
- **T2. Tampering with Data:** The credentials for ownership and access privileges are being attempted to be changed by malicious attackers.
- **T3. Repudiation:** Declining to engage in a specific action, such as denying the revocation of credentials or access requests, exemplifies the act of rejecting participation.
- **T4. Information Disclosure:** In the event of a data breach due to unauthorized access, formerly private user information may become publicly available.
- **T5. Denial of Service (DoS):** Malicious attempts to cause system availability to be disrupted by sending out excessive amounts of requests.
- **T6. Elevation of Privilege:** Illegal access aims to get elevated rights, with the ability to manipulate the decentralized identity system.

4.2.2 Requirement Analysis

Functional Requirements (FR) These are the requirements that are listed below.

- **F1.** Users should be able to independently generate and maintain their DIDs. It is imperative that the system has the ability to register, update, and revoke DIDs.
- **F2.** The system needs to be able to verify ownership credentials.
- **F3.** Camera owners must have the ability to grant, modify, and revoke capability credentials.
- **F4.** The system must enable users to selectively disclose information.
- **F5.** It should be possible for owners of cameras to rapidly withdraw their credentials and terminate their access privileges in the event that it becomes required.

Security Requirements (SR)
To address the revealed threats to personal safety, we examine a set of security specifications.

- **S1.** Strong authentication mechanisms are needed to avoid identity faking. Multi-factor authentication is one example. This approach will mitigate the threat of T1.
- **S2.** The use of strong encryption, secure communication methods, and regularly check for unauthorized data modifications will handle the threat of T2.

- **S3.** To reduce repudiation, the system should provide robust logging and non-repudiation capabilities to correctly monitor and evaluate user behaviors in order to mitigate the threat of T3.
- **S4.** To prevent unauthorized data exposure, the system must have strict access controls, encryption, and user education on responsible disclosure. It assures us to mitigate the threat of T4.
- **S5.** Rate limitation, traffic filtering, and scalable infrastructure can mitigate denial of service assaults which handle the threat of T5.
- **S6.** Security measures include least privilege, regular audits and access limit updates, and secure decentralized identity management. This approach enables to mitigate the threat of T6.

4.2.3 Architecture

Figure 4.1 shows the architecture design of smart home security system.The homeowner uses their SSI Wallet to generate a DID. The SSI Wallet communicates with the DID Registry to create and register a DID. This DID is a unique identifier that represents the homeowner's digital identity. The homeowner creates Ownership Credentials to their security cameras. These credentials prove ownership and are stored in the homeowner's SSI Wallet. The tenant uses their SSI Wallet to generate a DID. The tenant's SSI Wallet communicates with the DID Registry to create and register their DID. The registry records the tenant's DID, ensuring it is discoverable and verifiable. The owner will store those verifiable credential to the ledger's chaincode. The tenant will request for accessing the camera to the homeowner. The homeowner will ask for proofing credential from tenant's wallet. The tenant will share the credential proof from wallet. The homeowner verifies the tenant's DID. The homeowner checks the DID against the DID Registry to ensure it is valid. The homeowner uses the private Blockchain Network to issue Capability Credentials. The homeowner interacts with ledger to issue the Capability Credentials, specifying the tenant's access rights. The Capability Credentials are stored on the Blockchain Network. The credentials are stored immutably on the blockchain, ensuring they can be verified later. The SSI Wallet, which securely stores the tenant's digital credentials, pulls up the Capability Credentials that were issued by the homeowner. These credentials contain information about the tenant's access rights. Finally the owner approve access based on access policy.

From Figure 4.1, there are six major entities: homeowner, tenants, wallet, security camera. blockchain and SSI. The solid arrows represent the connections between two entities.

Decentralized Identifiers

The interpretation of the term "digital identity" is heavily influenced by the specific context in which it is employed.

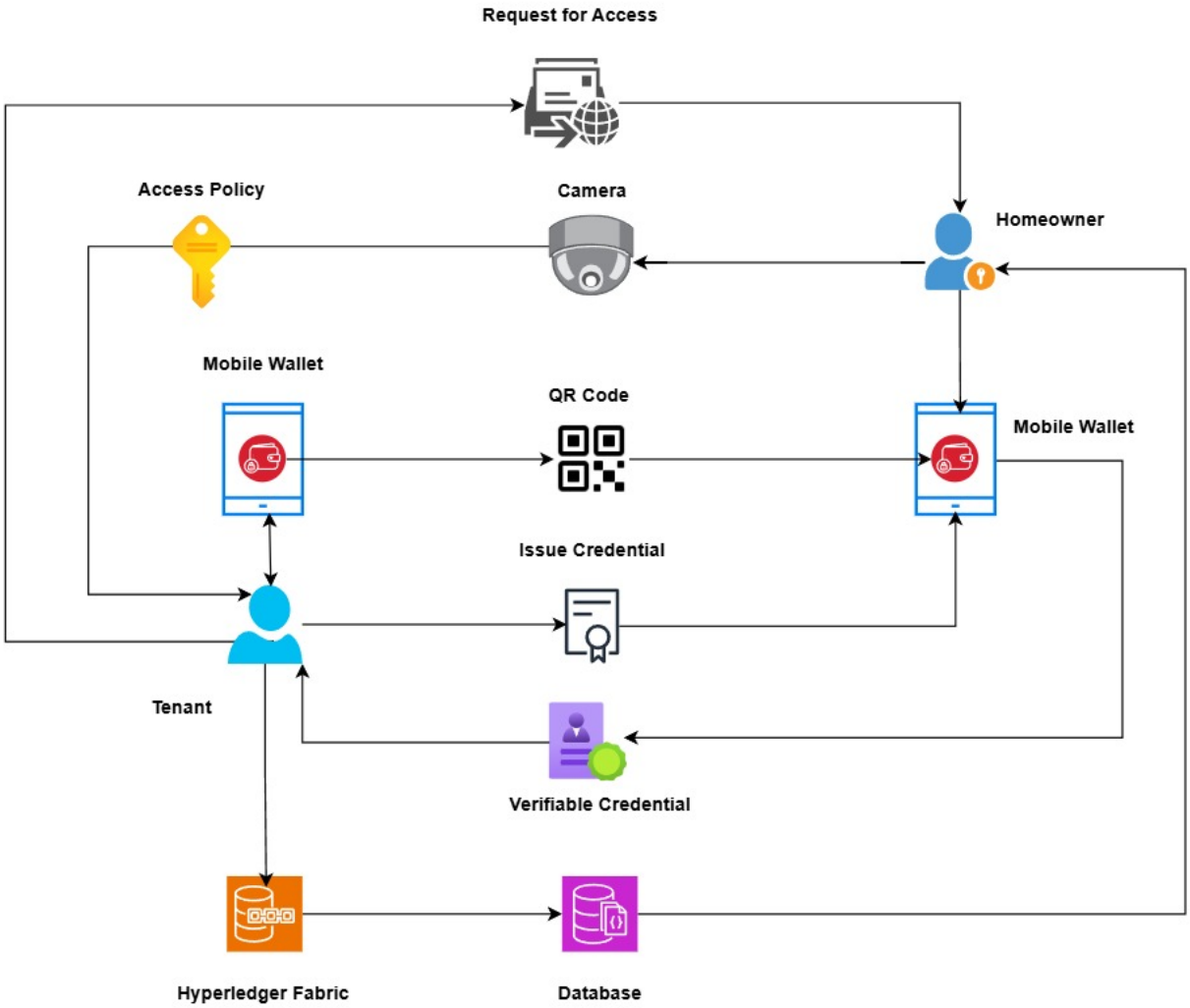

Figure 4.1: Architecture Design of Smart Home Security System

The phrase "KYC - Know your customer" is generally used by government stakeholders and regulated firms to refer to the real-world identity of a person in the context of identification operations. Primarily, a credential refers to an official document issued by a government body to a citizen, such as a passport, ID-card, or driver's license. A credential is a collection of traits that describe and distinguish one person from another, allowing for the identification of an individual.[14] In an online context, the identification of a person relies heavily on the eID-system, which requires a minimum of two authentication factors.

In the context of biometrics, inheritance refers to the automatic identification or comparable processes that are assisted by facial recognition technology, together with the inclusion of liveness checks. Knowledge refers to information that is used for authentication purposes, such as PINs, passwords, or one-time factors like SMS codes. Ownership: Smart cards, such as the eID-card, or secure elements on cellphones.

Although certain governments acknowledge that digital identity encompasses more than just a single certificate, their understanding is mostly confined to traits and credentials in a broader sense.

Mobile Wallet

Mobile wallet is like a virtual wallet that stays in a person's mobile device (e.g. a cell phone). It can store various types of information about various types of cards of an user. It is used for easy and quick payments through online transactions. Just like mobile wallets, SSI digital wallets are secure digital vaults for holding, managing sharing our Decentralized Identifiers (DIDs).[9] These wallets contain verifiable credentials such as, our education qualifications, professional certification or any other information that is relevant. These types of wallets are easily accessible, secure, private and instantly verifiable.[7]

Capability Credentials

A credential is any document, object, or data structure that, through a process of trust and authentication, attests to the identity of an individual.Credentialing is the process of evaluating an individual's knowledge, skill, or performance level in order to grant them a designation, such as a certificate or license [9].Credentials may encompass academic diplomas, academic degrees, certifications, security clearances, identification documents, insignia, passwords, user names, keys, powers of attorney, and so forth.Professional capability standards-based credentials are a solution to a problem that many individuals are still struggling to comprehend [9]. The landscape of work and learning has been eternally altered by digital disruption and globalization.[9] The acquisition of skills and knowledge is being transformed by this disruption.Capability credentials are a crucial element that facilitate secure , decentralized and user controlled access management.They Have the potential to utilize the capabilities of self sovereign identity frameworks and blockchain technology to offer a growth and long lasting solution for modern security requirement .[12]

Chapter 5

Use Case and Protocol Flow

In this section, we will describe the use cases and protocol flow that will demonstrate the interaction between various components of our system.

In 5.1, we present all the mathematical symbols and their corresponding notations.

Notation	**Description**
H_O	Homeowner
T	Tenant
DID	Decentralized Identifier Registry
DID_{H_O}	Decentralized Identifier of Homeowner
DID_T	Decentralized Identifier of Tenant
K_{H_O}	Public Key of Homeowner
$K_{H_O}^{-1}$	Private Key of Homeowner
K_T	Public Key of Tenant
K_T^{-1}	Private Key of Tenant
VC_{H_O}	Ownership Credentials issued to Homeowner
VC_T	Capability Credentials issued to Tenant
N_i	Fresh Nonce
$\{M\}_K$	Encryption of message M with key K
$\{M\}_K^{-1}$	Signature of message M with private key K^{-1}
$H(M)$	SHA-512 hash of message M
$\{...\}_K$	Communication over a channel encrypted with key K
$...K$	Optional data

Table 5.1: Cryptographic notations and their descriptions

Data Model

Notations	Descriptions
req = {serviceReq, inviteReq, credReq, proofReq, acceptReq}	Request types: service, invitation, credential, proof, accept
resp = {serviceResp, inviteResp[1–2], credResp, proofResp, acceptResp}	Response types: service, invitation, credential, proof, accept
serviceReq = {url, DID_T}	Service request URL with Tenant DID
inviteReq = {url, N_i}	Invitation request with URL and nonce
inviteResp1 = {N_i, DID_H_O}	First invitation response with nonce and home-owner DID
inviteResp2 = {N_i, DID_T}	Second invitation response with nonce and tenant DID
credReq = {"Requesting Credential"}	Credential request message
attrReq = {a1, a2, ..., an}	Attribute request for credentials
attrResp = {a1, av1}, {a2, av2}, ..., {an, avn}	Attribute response with name-value pairs
issueCapReq = {DID_T, accessRights}	Request to issue capability credentials
issueCapResp = {V_C_T}	Response with issued capability credentials
presentCredReq = {V_C_T}	Request to present capability credentials
verifyCredReq = {V_C_T}	Request to verify capability credentials
verifyCredResp = {status}	Response with verification status of capability credentials
acceptReq = {DID_T, securityCameraID}	Accept the request for camera
acceptResp = {"Accept Confirmation"}	Accept response message

Table 5.2: Data Model Table: Notations and Descriptions

Use Case 1:

Step	Message	Description
M1	$H_O \rightarrow SecurityCamera$:	Homeowner configures Security Camera to connect
M2	$H_O \rightarrow$ DID Registry :	Generate DID for Homeowner
M3	DID Registry $\rightarrow H_O$:	Return DID to Homeowner

Table 5.3: Ownership Credentials to Homeowner

Homeowner creates ownership to security cameras and stores them in SSI Wallet. Homeowner requests a DID by sending generateDIDReq(N_i) to the DID Registry. DID Registry returns DID to Homeowner using generateDIDResp(DID_{H_O}).

Use Case 2:

Step	Message	Description
M1	$T \rightarrow H_o$:	Tenant connects with Homeowner
M2	$T \rightarrow$ DIDRegistry :	Generate DID for Tenant
M3	DIDRegistry $\rightarrow T$:	Return DID to Tenant
M4	$T \rightarrow H_o$:	Tenant issues his credentials
M5	$H_O \rightarrow$ DID Registry :	Homeowner verifies Tenant's DID
M6	DID Registry $\rightarrow H_O$:	DID Registry confirms validity
M7	$H_O \rightarrow$ Blockchain :	Homeowner issues Capability Credentials via private blockchain
M8	$H_O \rightarrow$ Blockchain :	Blockchain stores Capability Credentials

Table 5.4: Establishing Connection Between Homeowner and Tenant

Tenant connects with Homeowner using SSI. Tenant generates a DID by sending a request to the DID Registry using the `generateDIDReq`(N_i) function. The DID Registry handles the request and provides the Tenant with a distinct DID. Tenant issues credentials using the `serviceReq`(url, DID_T) function to the Homeowner. Homeowner verifies Tenant's DID by submitting a verification request to the DID Registry using the `verifyDIDReq`(DID_T) function. DID Registry confirms the validity and responds with `verifyDIDResp`($status$) to the Homeowner. The homeowner issues Capability Credentials via the private blockchain using `issueCapReq`(DID_T, `accessRights`) to the blockchain. Blockchain stores Capability Credentials.

Use Case 3:

Step	Message	Description
M1	$T \rightarrow H_o$:	Tenant ask to access the camera from Homeowner
M2	$H_o \rightarrow T$:	Homeowner ask for proof of credential to Tenant
M3	$T \rightarrow H_o$:	Tenant presents Capability Credentials to Homeowner
M4	$H_o \rightarrow$ Blockchain :	Homeowner verifies the proof of Credentials
M5	Blockchain $\rightarrow H_o$:	Blockchain confirms validity of Capability Credentials
M6	$H_O \rightarrow$ T :	Homeowner access/deny based on valid credentials

Table 5.5: Requesting to Access the Camera by Tenant and Verifyng the Request

Tenant asks to access the camera from homeowner. This is the tenant's access request to the security system. The homeowner asks for proof of credential to the tenant. Tenant shares Capability Credentials to Homeowner. The homeowner verifies the proof of credentials. The Blockchain checks whether the provided credentials are valid. The Blockchain responds back to the homeowner with the verification status using `verifyCredResp (status)`. This informs the security system whether the credentials presented by the tenant are valid or not. Based on the validity of the credentials, the security cameras grant or deny access to the tenant.

Use Case 4:

Step	Message	Description
M1	$H_O \rightarrow$ T :	Homeowner accept requests to Tenant's access for security camera
M2	$H_O \rightarrow$ T :	Homeowner set access policy rule for tenant's access to security camera
M3	$T \rightarrow$ Security Camera :	Tenant will able to view the security camera

Table 5.6: Approval to Access the Camera

> The owner accepts requests for the tenant's access to the security camera `acceptResp (DIDT securityCamera)` to the blockchain. Homeowner set access policy rule for tenant to access the security camera. Finally, the tenant will see the live video feed of the camera.

In Figure 5.1, we demonstrate the flow of interactions between various components when the tenant intends to use the camera by renting from the homeowner. This is the process of transferring ownership from the initial owner to the subsequent tenant .

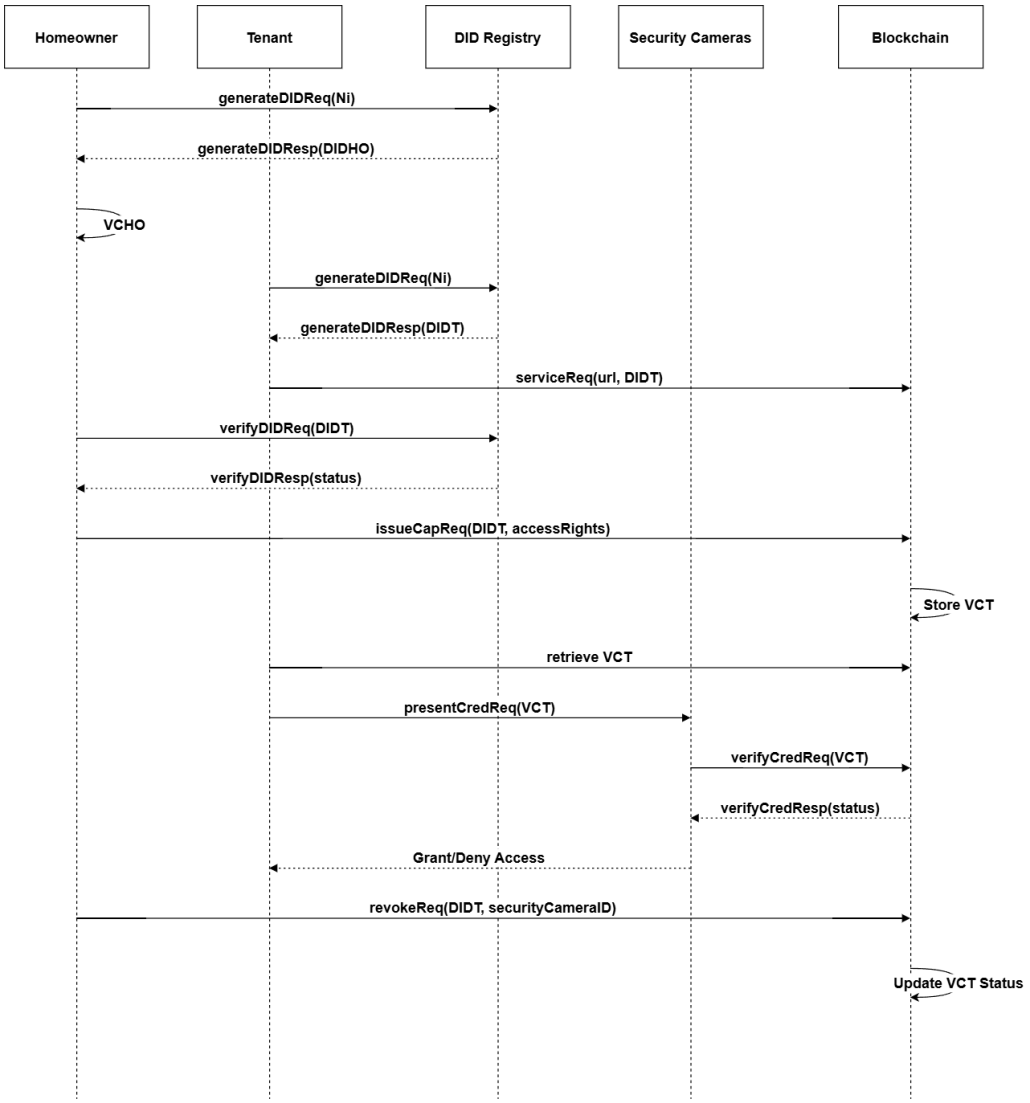

Figure 5.1: Protocol Flow of Smart Home Security System

Chapter 6

Implementation

Steps in the Smart Home System

1. The tenant establishes a connection with the homeowner through the SSI (Self-Sovereign Identity) system. Initially, the tenant requests a connection with the homeowner. After the request, the tenant sees a user interface (UI) containing a QR code.

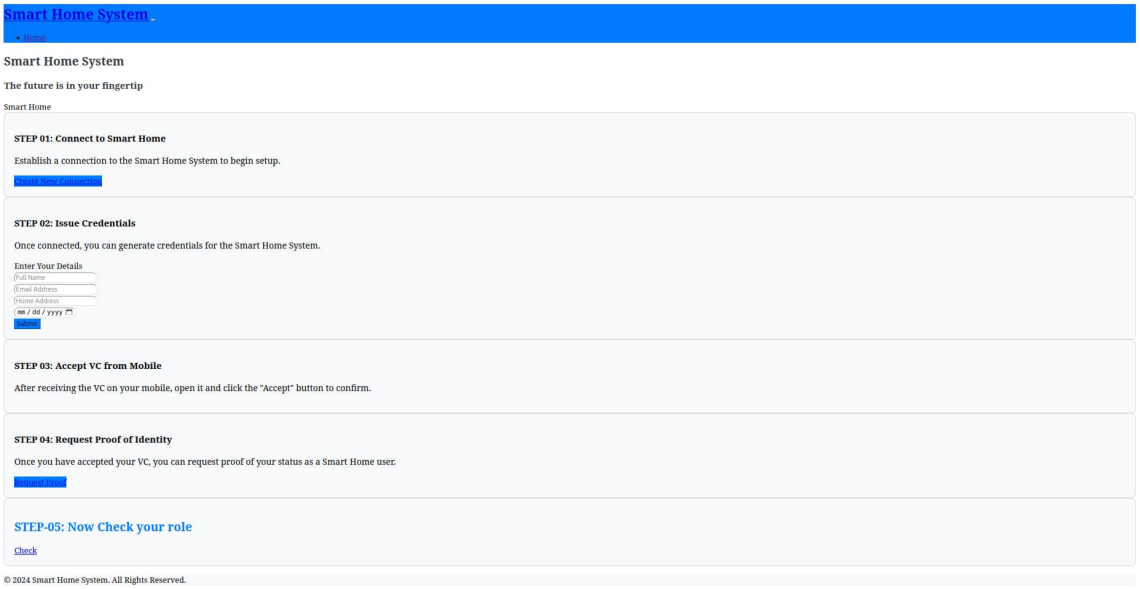

Figure 6.1: QR Code displayed in the UI for tenant connection.

2. The tenant uses the BC Wallet installed on their mobile device. The tenant unlocks their wallet using the pin they first created during the application's installation process.

3. After unlocking the wallet, the tenant selects the *Scan* option in the wallet. The tenant scans the QR code displayed in the interface, and the connection with the homeowner is established.

4. Next, the tenant issues their credentials to the homeowner for verification. After verifying the credentials, the verifiable credentials are sent to the tenant's wallet. The tenant then clicks the *Accept* button.

Figure 6.2: Tenant scanning the QR code to establish connection.

5. The tenant then requests proof of their status. The status is sent to the wallet, and the tenant accepts the role.

6. The tenant checks their role to confirm that they are listed as a tenant in the Smart Home System.

7. The tenant is presented with another Hyperledger form. The tenant fills out their credentials and specifies the amount of time they wish to have access to the camera. After submitting the form, the data is stored in both CouchDB and MongoDB. The tenant is asked to store their tenant ID.

8. Finally, the tenant requests camera access by providing the tenant ID they created when submitting the form. The tenant sends the tenant ID to the homeowner to request access. The homeowner can view the tenant's data, including the requested access time.

9. The homeowner reviews the request and accepts it. A notification is sent to the tenant, who is now able to view the camera for the specified duration. If the homeowner declines the request, the tenant is also notified.

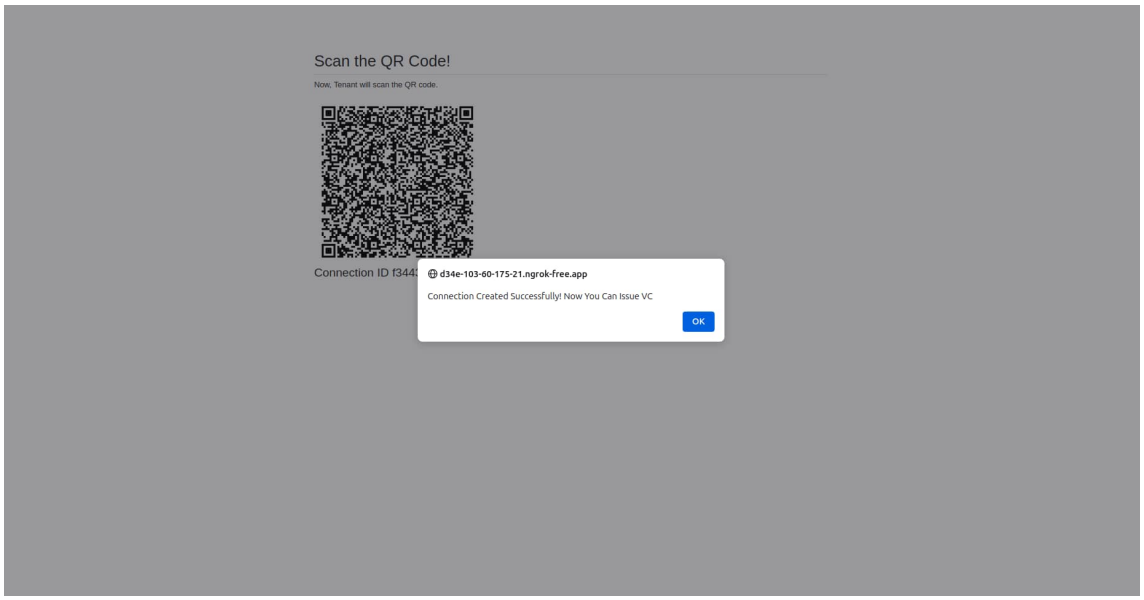

Figure 6.3: QR code scan confirmation.

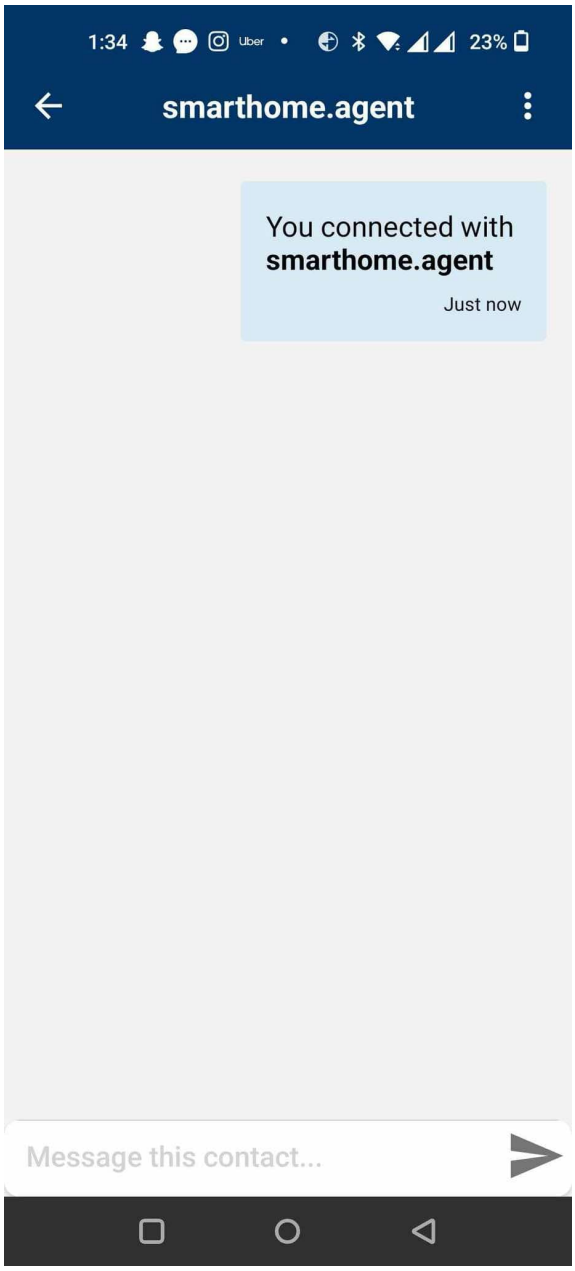

Figure 6.4: Connection established between tenant and homeowner.

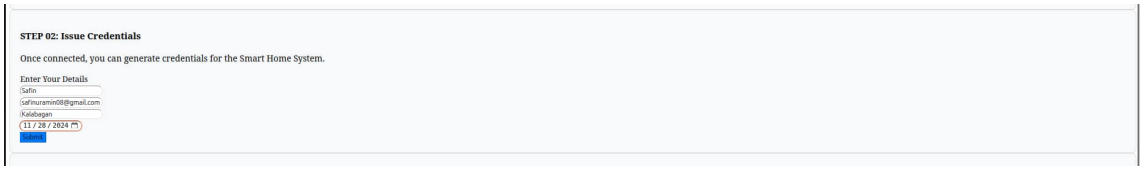

Figure 6.5: Tenant submits credentials for verification.

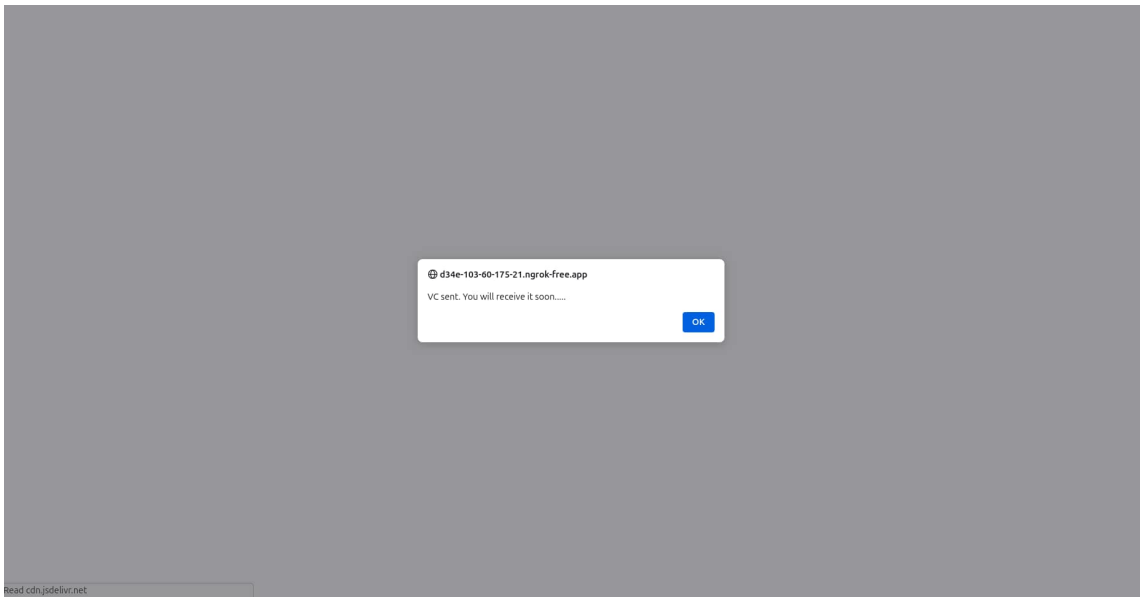

Figure 6.6: Verification of credentials completed.

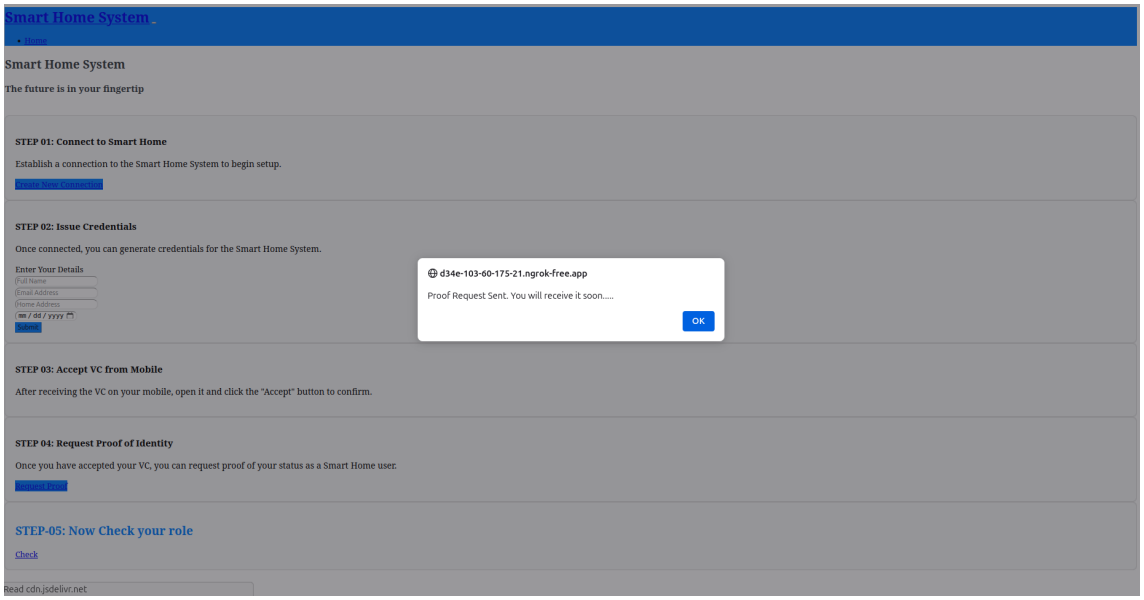

Figure 6.7: Verified credentials received in tenant's wallet.

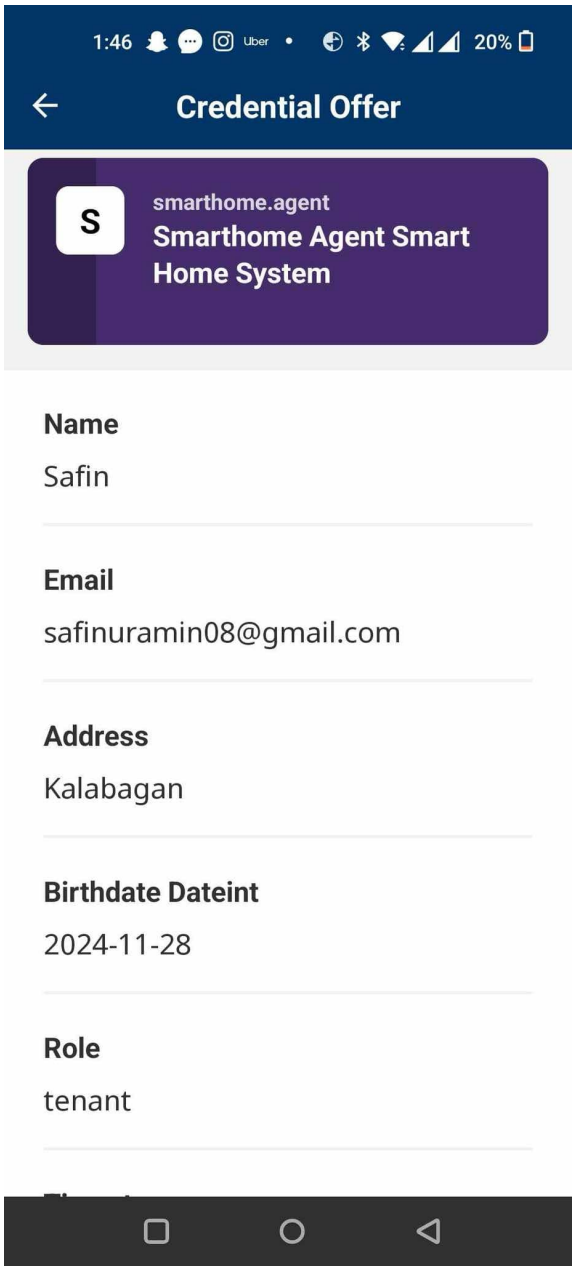

Figure 6.8: Tenant clicks 'Accept' to confirm credentials.

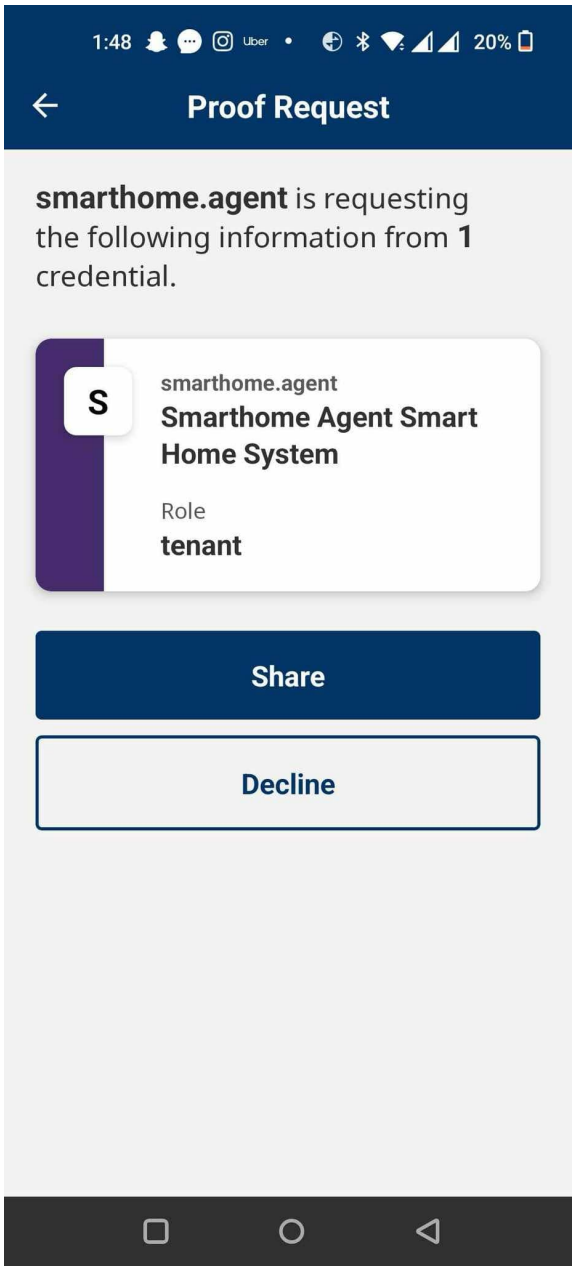

Figure 6.9: Tenant requests and receives proof of their status.

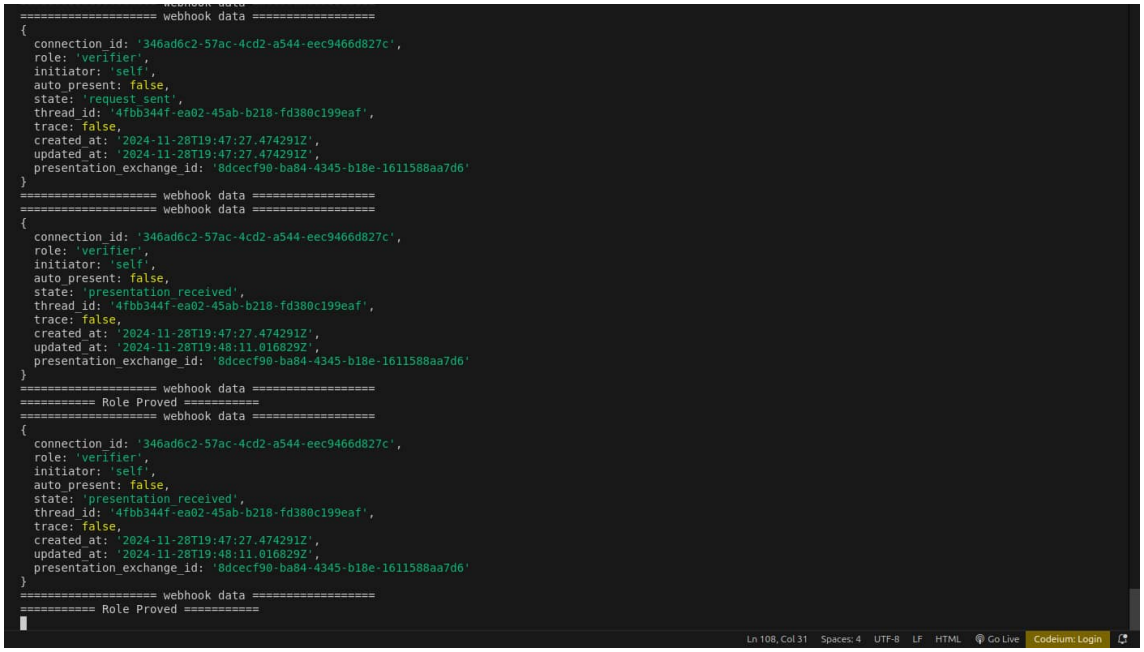

Figure 6.10: Offer is send for issuing VC

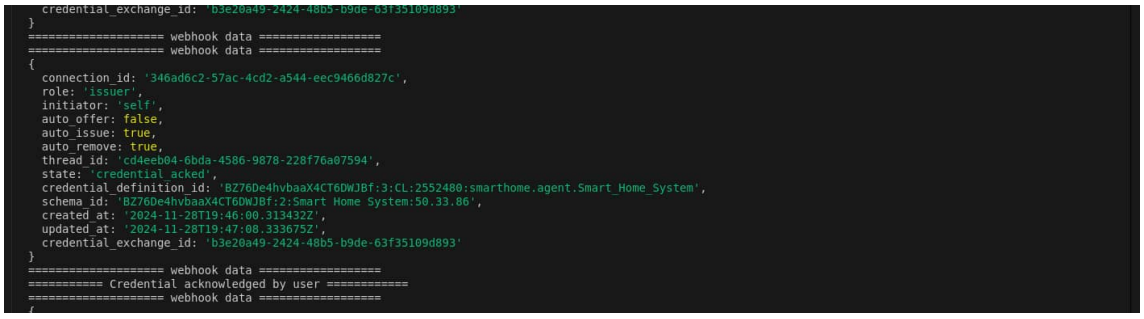

Figure 6.11: Tenant's role confirmed.

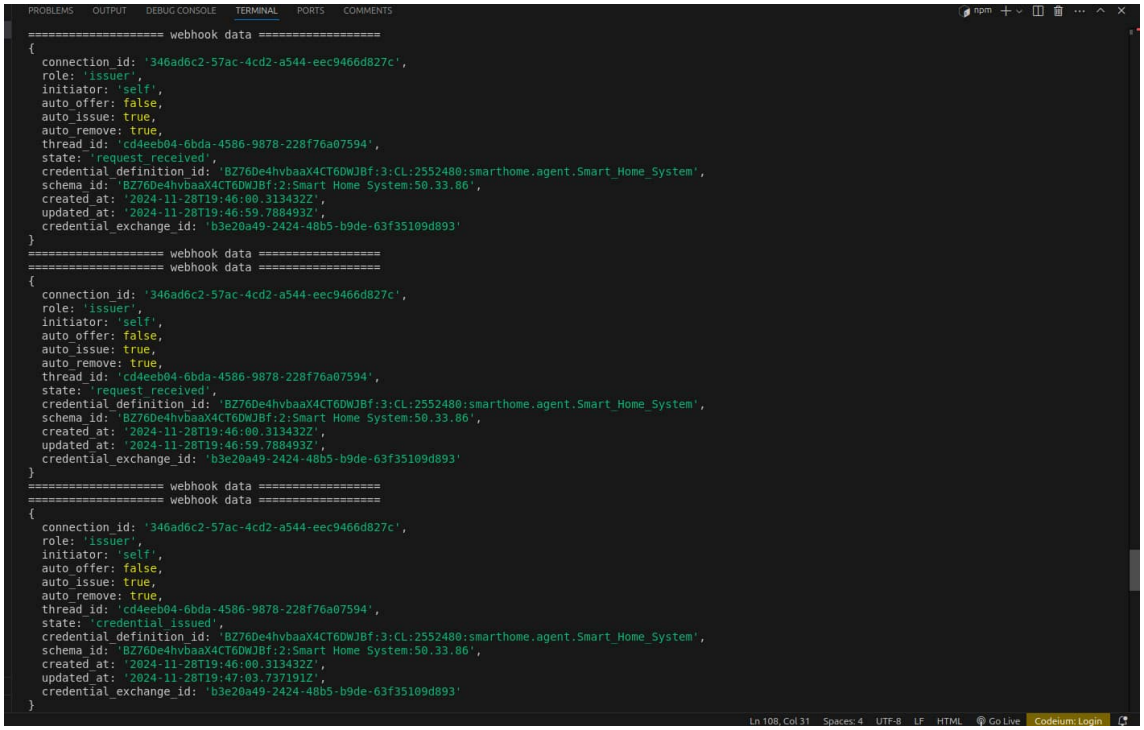

Figure 6.12: Role check in the Smart Home system.

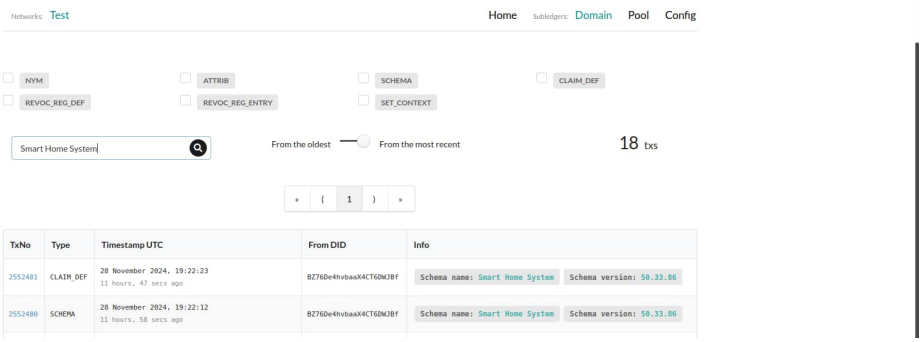

Figure 6.13: Further role verification in the system.

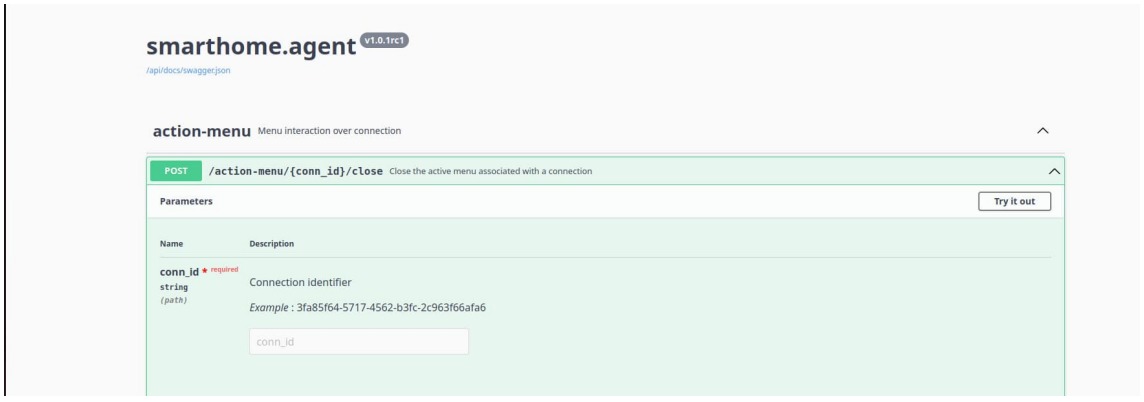

Figure 6.14: Role confirmed as tenant in the system.

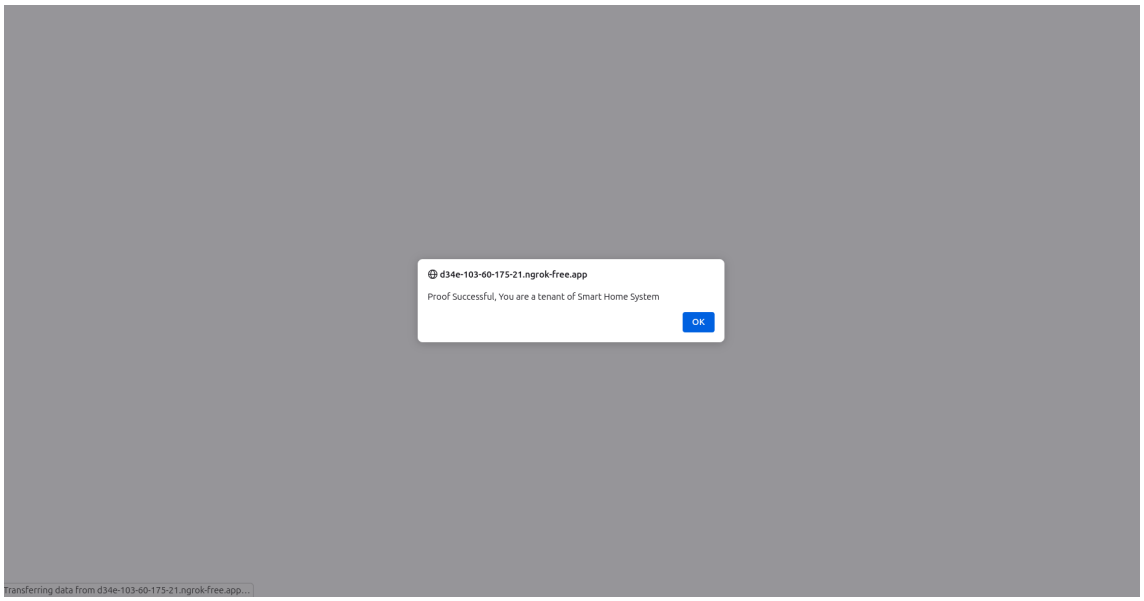

Figure 6.15: Final confirmation of tenant status.

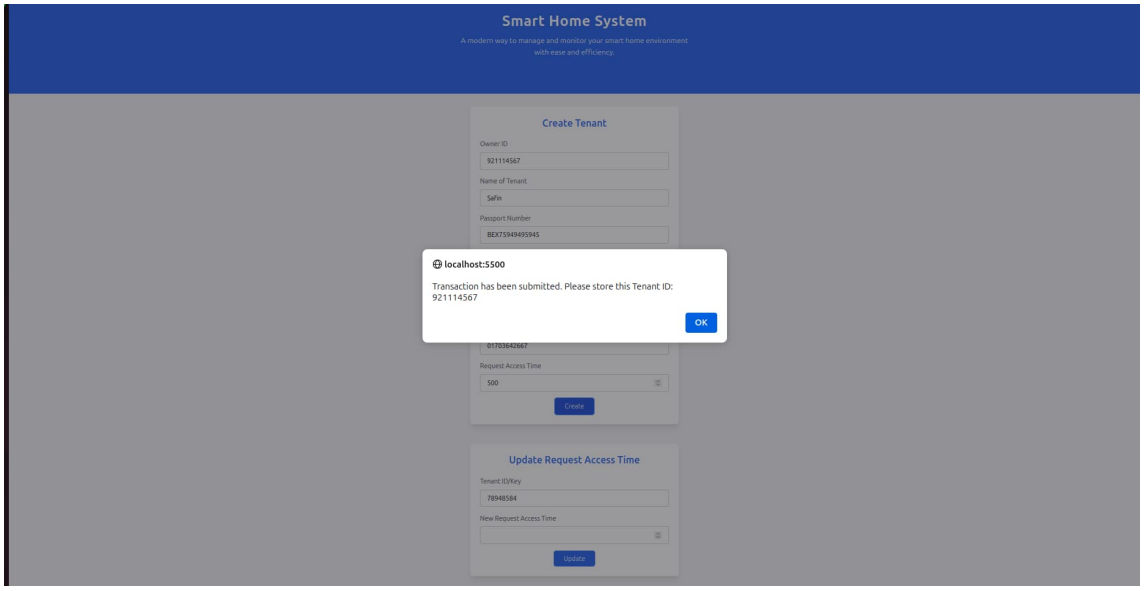

Figure 6.16: Tenant is reminded to store their tenant ID

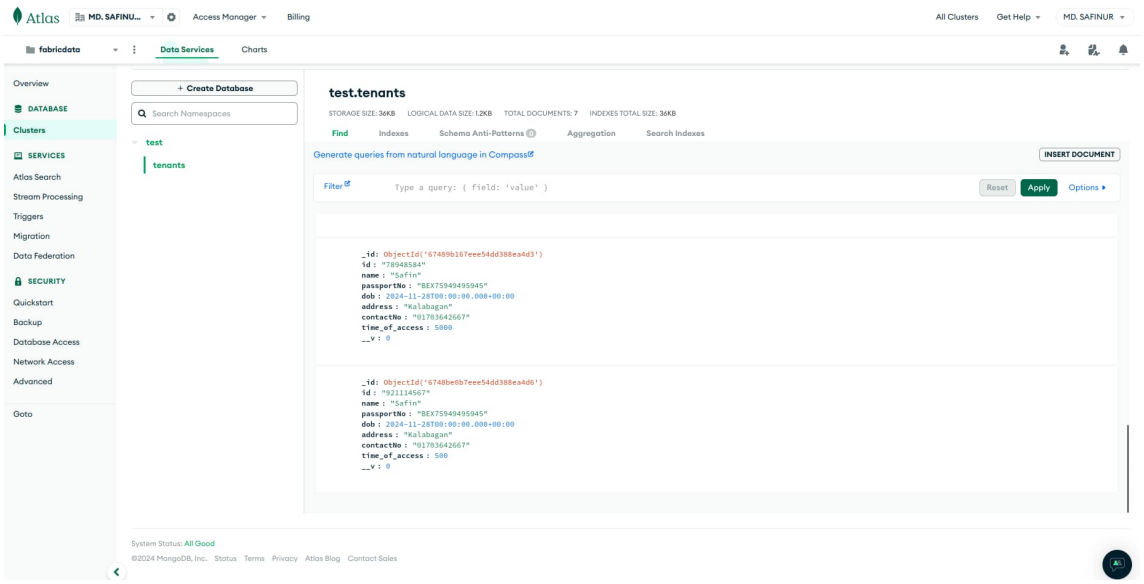

Figure 6.17: Data is stored in MongoDB

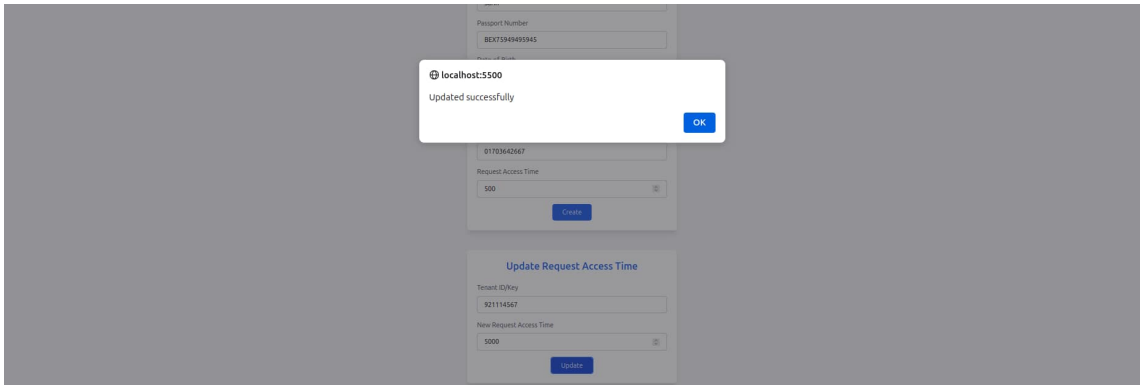

Figure 6.18: Tenant updates his access time

Figure 6.19: Tenant Data is updated on MongoDB

```
Connected to MongoDB server
Wallet path: /home/safin/fabric/fabric-samples/fabcar/javascript/wallet
Transaction has been submitted. Please store this DID: 921114567
Tenant data saved to MongoDB
Wallet path: /home/safin/fabric/fabric-samples/fabcar/javascript/wallet
Change Request time for access Transaction has been submitted
Tenant 921114567 updated in MongoDB with new time_of_access: 5000
```

Figure 6.20: Confirmation of data submission.

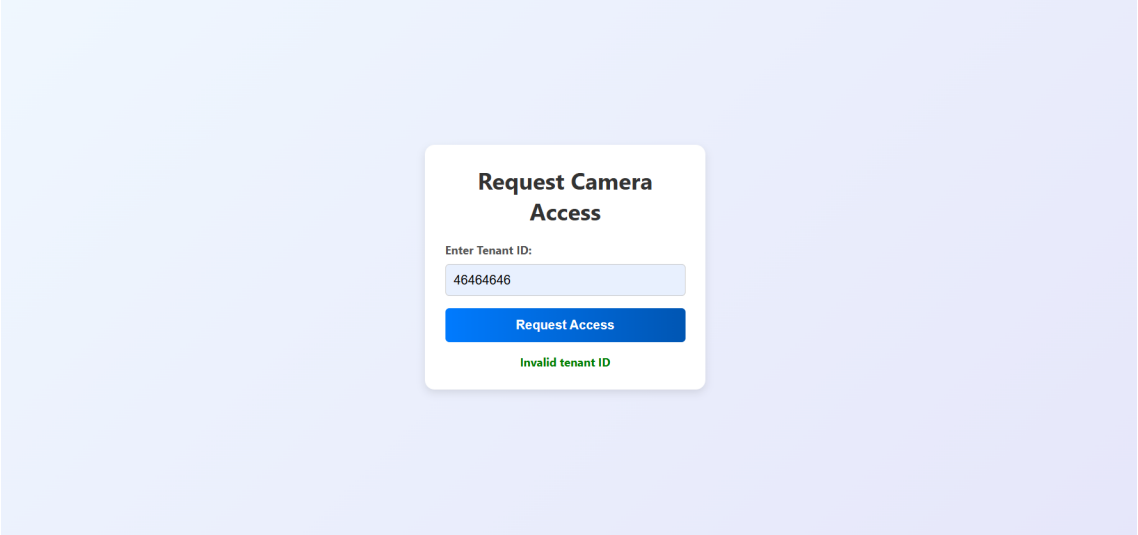

Figure 6.21: Invalid ID access request.

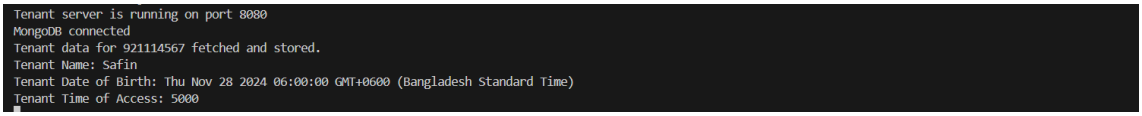

Figure 6.22: Homeowner receives request from the tenant and extracts data from database.

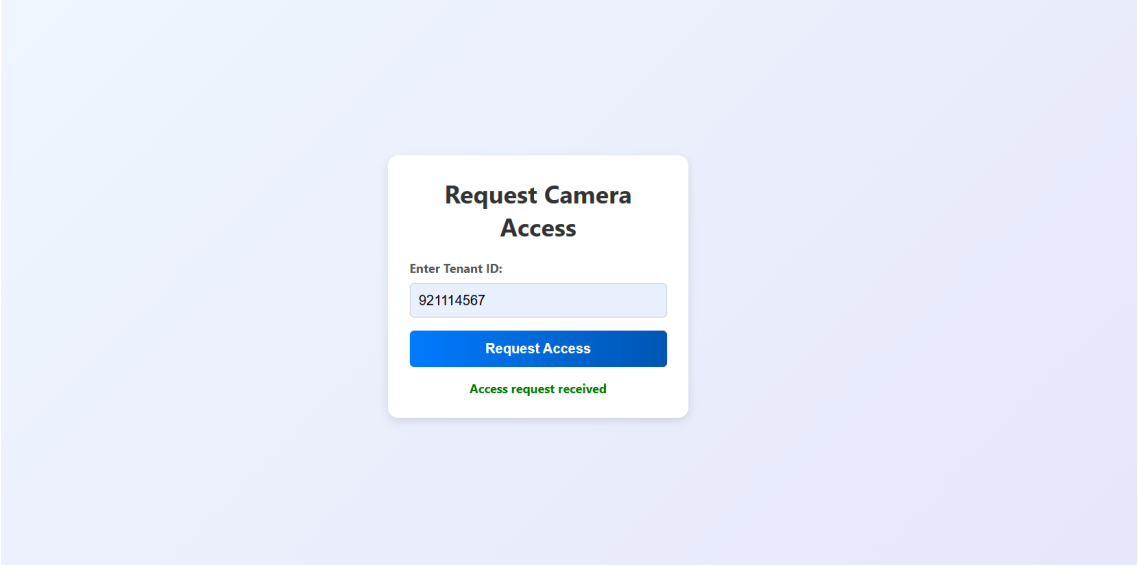

Figure 6.23: Tenant sends Homeowner for the camera access request.

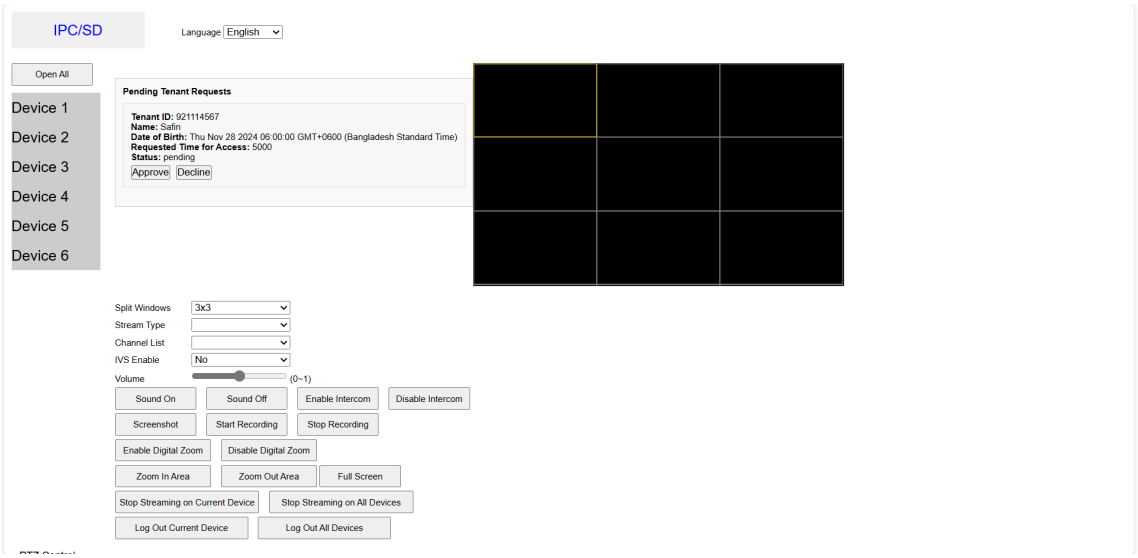

Figure 6.24: Homeowner reviews the request of Tenant

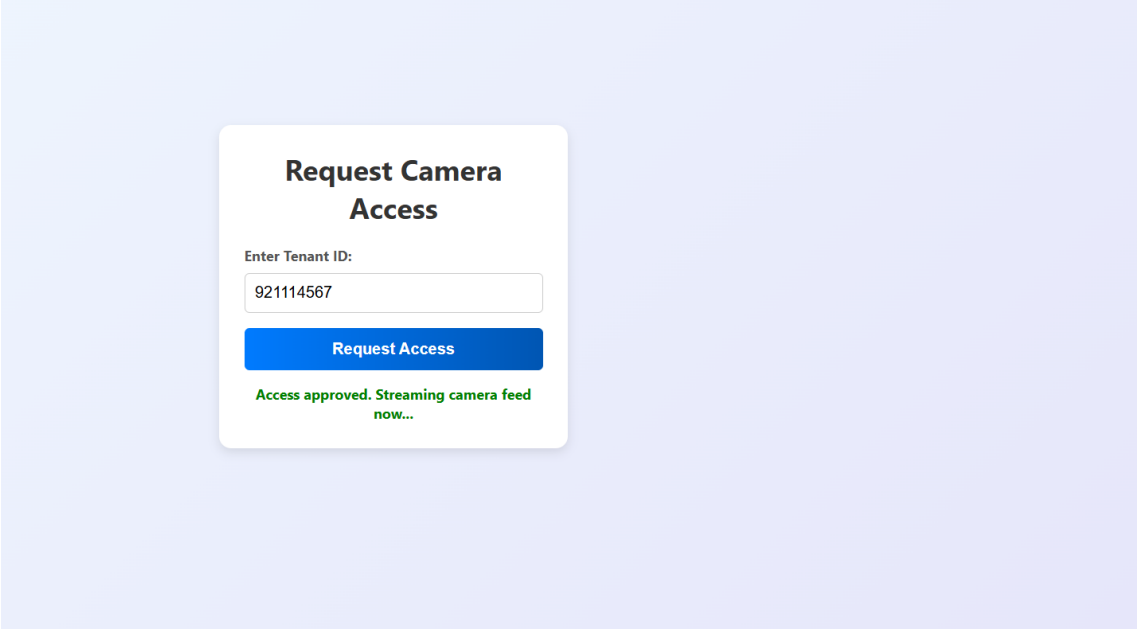

Figure 6.25: Homeowner approves the request.

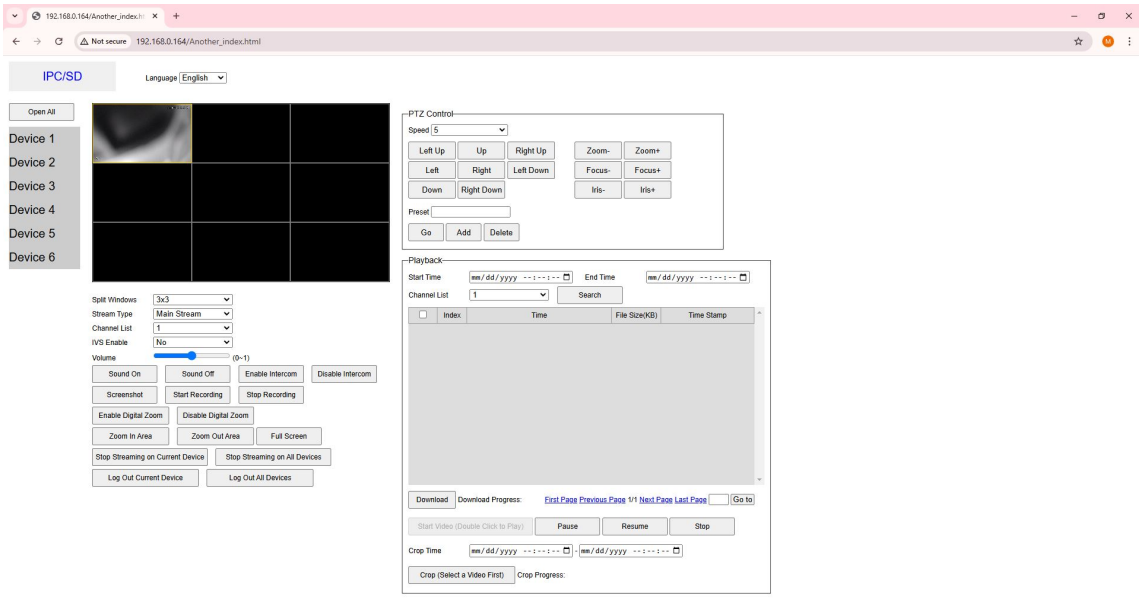

Figure 6.26: Tenant is able to view the camera with the access time.

Chapter 7

Discussions

7.1 Advantages

1. Independent Generation and Maintenance of Decentralized Identifiers (DIDs)

Tenants are empowered to independently generate and manage their own Decentralized Identifiers (DIDs). This ensures that tenants have full control over their digital identities, reducing reliance on centralized authorities and enhancing privacy and autonomy in digital transactions.

2. Strong Authentication and Encryption via Self-Sovereign Identity (SSI)

The system leverages Self-Sovereign Identity (SSI) technology to authenticate tenants and ensure the confidentiality of their credentials. SSI allows for secure, decentralized identity verification without the need for intermediaries, relying on advanced cryptographic methods to protect data and ensure secure transactions.

3. Granular Access Control (Granting, Modifying, and Revoking Capabilities)

The system provides homeowners with robust access control features. Homeowners can grant, modify, or revoke access to certain capabilities, ensuring that tenant interactions with the property and system are tightly regulated. This flexibility allows homeowners to maintain control over the access rights associated with their property.

4. Homeowner Consent and Request Rejection

Tenant requests can be reviewed and rejected by homeowners, who can make sure that no action is performed without the tenant's express consent. Because homeowners have complete control over the requests and permissions given to tenants, this feature improves security and confidence between the two parties.

5. Data Integrity through Hyperledger Fabric Database

The Hyperledger Fabric blockchain is used by the system to store and protect data, guarding against unwanted modifications and manipulation. The permissioned and immutable ledger offered by Hyperledger Fabric ensures the integrity of data records and makes it challenging for bad actors to change information covertly.

6. Prevention of Spoofing via Unique Tenant ID

Each tenant is assigned a unique Tenant ID, ensuring that their identity is distinct and cannot be spoofed by unauthorized entities. This uniqueness strengthens the security of the system by preventing impersonation or fraudulent access attempts.

7. Prevention of Denial-of-Service (DoS) Attacks

To ensure that their identities are different and impossible for unauthorized parties to impersonate, each renter is given a unique renter ID. Because of its uniqueness, the system is more secure against fraudulent access attempts and impersonation.

8. Tenant Data Verification and Transparency for Homeowners

Tenant data is accessible to and verifiable by homeowners. By doing this, the system is made transparent, enabling homeowners to verify the accuracy of the data and decide whether to approve requests or give authorization.
These advantages collectively enhance the system's security, privacy, and trustworthiness, creating a more efficient and reliable platform for tenant-homeowner interactions.

7.2 Limitations

The challenges faced while managing two different networks, the Blockchain network and the Internet of Things network were a major obstacle to the Smart Home System's deployment. Because it took a lot of optimization and careful configuration to ensure real-time communication between various networks, synchronization was a major problem. Request processing and access authorization verification were delayed as a result of error caused by the intrinsic variations in network design and functionality.The inability of IoT devices to manage the extra computational and storage overhead brought about by blockchain operations was another drawback. These gadgets frequently don't have the processing capacity to do jobs like QR code scanning, credential verification, and blockchain interaction effectively. In certain instances, this caused the system to become less responsive.The inability of IoT devices to manage the extra computational and storage overhead brought about by blockchain operations was another drawback. These gadgets frequently don't have the processing capacity to do jobs like QR code scanning, credential verification, and blockchain interaction effectively. In certain instances, this caused the system to become less responsive.The inability of IoT devices to manage the extra computational and storage overhead brought about by blockchain operations was another

drawback. These gadgets frequently don't have the processing capacity to do jobs like QR code scanning, credential verification, and blockchain interaction effectively. In certain instances, this caused the system to become less responsive.
The inability of IoT devices to manage the extra computational and storage overhead brought about by blockchain operations was another drawback. These gadgets frequently don't have the processing capacity to do jobs like QR code scanning, credential verification, and blockchain interaction effectively. In certain instances, this caused the system to become less responsive.
Furthermore, the system became more sophisticated when SSI and the Capability-Based Access Model were integrated. It took a lot of work to retain lightweight solutions for IoT environments while guaranteeing smooth compatibility between these technologies. It took a lot of effort to debug and troubleshoot network interactions, especially while testing the system in real-time circumstances.

Lastly, the controlled setting in which the prototype was evaluated failed to adequately represent the unpredictable nature of actual IoT networks. The system's scalability and dependability need to be confirmed through additional testing using bigger, more intricate configurations. Notwithstanding these drawbacks, the study shows how SSI and blockchain technology can be combined to enhance the security and effectiveness of smart home systems.

7.3 Future Work

Additionally, the integration of SSI and the Capability-Based Access Model advanced the system. Maintaining lightweight solutions for IoT contexts while ensuring seamless interoperability between different technologies required a great deal of effort. Debugging and troubleshooting network connections required a lot of work, particularly when testing the system under real-time conditions.
Finally, the controlled environment used to test the prototype does not accurately reflect the unpredictability of real-world IoT networks. Further testing with larger, more complex configurations is required to verify the system's scalability and reliability. Despite these limitations, the study demonstrates how SSI and blockchain technology can be used in tandem to improve the efficacy and security of smart home systems.

Chapter 8

Conclusion

Our proposed research system is an integration of Self-Sovereign Identity and Capability-Based Access Model into smart home security. To complete our proposed system, we are going to follow five steps which are threat modelling, requirement analysis, architecture design, protocol flow & implementation. We have successfully done five steps of our proposed system to fulfill our research objectives. In our system, we have successfully stored the user's data in a private blockchain which is hyperledger fabric so that data can't be altered easily. Before that, owner verifies data before storing to blockchain through SSI. When anyone wants to access a camera from the owner he needs to show proof. By validating it, the owner will set policy rules to access it. So, it ensures the security of the smart home system. Finally, we hope that it opens the doors for developers or researchers to create or develop better systems in this domain for future work.

Bibliography

[1] P. Samarati and S. C. de Vimercati, "Access control: Policies, models, and mechanisms," in *Foundations of Security Analysis and Design*, ser. Lecture notes in computer science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 137–196.

[2] S. Gusmeroli, S. Piccione, and D. Rotondi, "Iot access control issues: A capability based approach," Jul. 2012. DOI: 10.1109/IMIS.2012.38.

[3] A. Shostack, *Threat modeling*, en. Nashville, TN: John Wiley & Sons, Feb. 2014.

[4] A. Ouaddah, H. Mousannif, A. Abou Elkalam, and A. Ait Ouahman, "Access control in the internet of things: Big challenges and new opportunities," en, *Comput. Netw.*, vol. 112, pp. 237–262, Jan. 2017.

[5] D. Di Francesco Maesa, P. Mori, and L. Ricci, "Blockchain based access control services," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, pp. 1379–1386. DOI: 10.1109/Cybermatics_2018.2018.00237.

[6] R. Xu, Y. Chen, E. Blasch, and G. Chen, "BlendCAC: A smart contract enabled decentralized capability-based access control mechanism for the IoT," en, *Computers*, vol. 7, no. 3, p. 39, Jul. 2018.

[7] R. Belchior, B. Putz, G. Pernul, M. Correia, A. Vasconcelos, and S. Guerreiro, "Ssibac: Self-sovereign identity based access control," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2020, pp. 1935–1943. DOI: 10.1109/TrustCom50675.2020.00264.

[8] S. N, V. Bhat K, and M. Rajarajan, "Blockchain-based scheme for authentication and capability-based access control in iot environment," in *2020 11th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*, 2020, pp. 0323–0330. DOI: 10.1109/UEMCON51285.2020.9298116.

[9] N. Sivaselvan, W. Asif, B. K. Vivekananda, and M. Rajarajan, "Authentication and capability-based access control: An integrated approach for iot environment," in *2020 12th International Conference on Communication Software and Networks (ICCSN)*, 2020, pp. 110–117. DOI: 10.1109/ICCSN49894.2020.9139051.

[10] S. Terzi, C. Savvaidis, K. Votis, D. Tzovaras, and I. Stamelos, "Securing emission data of smart vehicles with blockchain and self-sovereign identities," in *2020 IEEE International Conference on Blockchain (Blockchain)*, 2020, pp. 462–469. DOI: 10.1109/Blockchain50366.2020.00067.

[11] Y. Zhang, B. Li, B. Liu, J. Wu, Y. Wang, and X. Yang, "An attribute-based collaborative access control scheme using blockchain for IoT devices," en, *Electronics (Basel)*, vol. 9, no. 2, p. 285, Feb. 2020.

[12] N. Fotiou, V. A. Siris, and G. C. Polyzos, "Capability-based access control for multi-tenant systems using oauth 2.0 and verifiable credentials," in *2021 International Conference on Computer Communications and Networks (ICCCN)*, IEEE, 2021, pp. 1–9.

[13] Y. Liu, Q. Lu, S. Chen, *et al.*, "Capability-based IoT access control using blockchain," en, *Digit. Commun. Netw.*, vol. 7, no. 4, pp. 463–469, Nov. 2021.

[14] B. Mohammed Amine, B. Xia, A. Abuassba, H. Ning, and Q. Lu, "Iot-ccac: A blockchain-based consortium capability access control approach for iot," *PeerJ Computer Science*, vol. 7, e455, Apr. 2021. DOI: 10.7717/peerj-cs.455.

[15] P. Rao and P. Saraswathi, "Securing internet of things devices by enabling ethereum blockchain using smart contracts," *Building Services Engineering Research and Technology*, vol. 43, Apr. 2022. DOI: 10.1177/01436244221078933.

[16] J. F., *The trust triangle as defined in the context of Self-Sovereign identity*, en, https://medium.com/iamx-own-your-identity/the-trust-triangle-as-defined-in-the-context-of-self-sovereign-identity-e876bc361cc8, Accessed: 2024-1-24, Mar. 2023.