electronics

MDPI

Article

Blockchain-Enabled Access Management System for Edge Computing

Yong Zhu [1,2,3,*], Chao Huang [1], Zhihui Hu [2], Abdullah Al-Dhelaan [4] and Mohammed Al-Dhelaan [4]

[1] School of Computer Engineering, Jinling Institute of Technology, Nanjing 211169, China; huangchao@jit.edu.cn
[2] School of Computer Science & Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; hzhnjupt@163.com
[3] College of Computer and Information Technology Engineering, Hohai University, Nanjing 210098, China
[4] Computer Science, King Saud University, Riyadh 11495, Saudi Arabia; dhelaan@ksu.edu.sa (A.A.-D.); mdhelaan@ksu.edu.sa (M.A.-D.)
* Correspondence: zhudz@jit.edu.cn; Tel.: +86-18168092326; Fax: 025-86188957

Abstract: In the post-cloud era, edge computing is a new computing paradigm with data processed at the edge of the network, which can process the data close to the end-user in real time and offload the cloud task intelligently. Meanwhile, the decentralization, tamper-proof and anonymity of blockchain technology can provide a new trusted computing environment for edge computing. However, it does raise considerable concerns of security, privacy, fault-tolerance and so on. For example, identity authentication and access control rely on third parties, heterogeneous devices and different vendors in IoT, leading to security and privacy risks, etc. How to combine the advantages of the two has become the highlight of academic research, especially the issue of secure resource management. Comprehensive security and privacy involve all aspects of platform, data, application and access control. In. this paper, the architecture and behavior of an Access Management System (AMS) in a proof of concept (PoC) prototype are proposed with a Color Petri Net (CPN) model. The two domains of blockchain and edge computing are organically connected by interfaces and interactions. The simulation of operation, activity and role association proves the feasibility and effectiveness of the AMS. The instances of platform business access control, data access control, database services, IOT hub service are run on Advantech WISE-PaaS through User Account and Authentication (UAA). Finally, fine-grained and distributed access control can be realized with the help of a blockchain attribute. Namely, smart contracts are used to register, broadcast, and revoke access authorization, as well as to create specific transactions to define access control policies.

Keywords: AMS; CPN; blockchain; edge computing; decentralized applications (DApps)

check for updates

Citation: Zhu, Y.; Huang, C.; Hu, Z.; Al-Dhelaan, A.; Al-Dhelaan, M. Blockchain-Enabled Access Management System for Edge Computing. *Electronics* **2021**, *10*, 1000. https://doi.org/10.3390/electronics10091000

Academic Editor: Nurul I. Sarkar

Received: 21 February 2021
Accepted: 17 April 2021
Published: 22 April 2021

1. Introduction

With the popularity of IoT and mobile computing, massive data generated by scattered terminal devices are stored on a large number of edge nodes, providing data sources for AI applications. However, traditional cloud-based architectures face many problems. The concept of edge computing emerged in 2017, which gave birth to the research direction and development of an AI cloud-edge system. Therefore, the new paradigm of decentralizing computing and storage to the edge has gradually become an important research interest [1–3].

Blockchain adopts an accounting technology that is jointly maintained by multiple parties to ensure the security of transmission and access with cryptography, and is able to achieve consistent data storage, difficult to tamper with, and prevent repudiation, namely distributed ledger technology. As the underlying technology supporting Bitcoin, blockchain can build ledger accounts in a secure and verifiable way in a pan-central environment. Its development has gone through three versions, respectively, represented by Bitcoin, smart contract and DApps.

The natural pan-central distributed trust characteristics of blockchain provide new ideas for designing edge computing frameworks and paradigms. The intelligent devices distributed in the edge nodes need to communicate frequently and cooperate to complete AI computing tasks or intelligent group decision-making. However, both the device itself and the communication between devices are faced with a variety of network security threats, such as the possibility of device failure or malicious attack, in which case the transmitted information may be leaked or tampered with [4]. As a kind of verifiable and tamper-proof ledger supported by cryptography, blockchain can guarantee information sharing and secure interaction in decentralized untrusted environments through transaction records, validity consensus and smart contracts, which play an important role in the scenario of edge computing. Therefore, the advantages of the integration of blockchain technology and edge computing are as follows:

- The application architecture of blockchain pan-center is more suitable for collaborative processing activities in edge computing scenarios;
- The security of data storage and transmission can be ensured by the non-tamper property of blockchain in edge computing;
- The blockchain mechanism of consensus and smart contracts can effectively stimulate the sharing and exchange of information and data in edge computing to improve and optimize the AI trust model [5].

By providing local computing power, edge computing enables blockchain deployment to support solving Proof of Work (PoW) puzzles, hashing, encryption algorithms and possibly consensus. Interactions for edge computing can be modeled as market activities to help the edge service providers achieve optimal resource management policy and profit. The service provider deploying the MEC service aims to maximize the profit through pricing. Therefore, the miners in the blockchain have to consider the reward from mining and the price paid to the provider in deciding on the service demand [6]. The PoW puzzle can be offloaded to the edge computing server, and the miners are priced by the provider. In practice, a similar concept that integrates edge computing and blockchain has been realized. For example, Microsoft provides Blockchain as a Service (BaaS) on the Azure cloud platform. A United Kingdom company CloudHashing offers Mining as a Service (MaaS) where the users only buy software services online to mine Bitcoins, without installing and deploying hardware equipment. IBM provides a Watson IoT platform to manage IoT data in a private blockchain ledger, which is integrated into IBM's business-level cloud services [7,8].

It has great potential to combine blockchain with edge computing and apply it as a solution in various scenarios [9]. It is an effective solution to build a secure and trusted edge computing framework based on blockchain properties of encryption and non-tamper. For example, VSNs [10] allow users to customize encryption policies for sensitive data, provide conditional access and decryption query methods, and perform various operations through protected smart contracts. UniqueID [11] proposes an identity as a service model, which improves the operation efficiency of connecting and controlling key devices, such as sensors, actuators, and devices in enterprises, and overcomes the scalability burden of traditional protocols. IBM Trusted Identity [12] is dedicated to creating secure, blockchain-enabled trusted identities for everyone on the Internet and extending identity management to edge network applications. It is also an effective method to manage edge computing data by using the technology of on-chain and software-hardware collaboration. In order to balance the contradiction between security and performance, mainstream hardware vendors, such as Intel's SGX, ARM's TrustZone, RISC-V's keystone, AMD's SME/SEV, have launched hardware security protection solutions represented by the Trusted Execution Environment (TEE) in recent years, providing a protected storage and execution environment security policy for data and operation on the chain. TEE combined with blockchain technology realizes a feasible data security protection scheme with fast speed and low cost [13–15].

Based on the above exposition about IoT, edge computing and blockchain, we can combine edge computing advantages of real-time processing close to the end-user and task migration with distributed security features of blockchain to build applications suitable for IoT scenarios. The objective of this paper to optimize their key technologies, build a decentralized and secure AMS model, and carry on design space exploration (DSE) and demonstration.

The key contributions of this article are as follows:

- By studying the working principle of IoT, edge computing and blockchain and their relationships, the system mode that can combine their advantages will be found;
- The security and reliability of AMS and the scalability of the system are guaranteed by referring to the blockchain native properties of tamper-proof and decentralization;
- The model of blockchain-enabled AMS for edge computing in CPN is built, and the main behavior of the system is simulated with CPN-tools. The feasibility of this objective is verified theoretically for the PoC. At the same time, the AMS in Advantech WISE-PaaS is used to give a practical verification.

This paper is organized as follows. After introducing the research significance, objectives and contributions in Sections 1 and 2 provide the related work about IoT, edge computing and blockchain. Section 3 presents the AMS protocol referenced by IETF and OMA, which suit IoT. The model of blockchain-enabled edge computing is put forward in Section 4, including interfaces, interactions, architecture and behavior. Section 5 discusses the simulation for AMS in CPN-Tools. Section 6 concludes the paper. Finally, future research problems are raised.

2. Related Work

Blockchain is composed of a block and chain. The block is a permanent data storage unit for recording transactions, which is chained in accordance with time stamps in chronological order, maintained jointly by nodes in the network through a consensus mechanism to form a distributed non-tampered ledger. In short, it is an open and distributed database, that is, a digital ledger, which can effectively record transactions between parties in a verifiable and non-tamper-proof way. More broadly, blockchain can also serve as an underlying framework to provide a feasible solution to enhance the credibility and security of the system, which is widely used in the financial economy, IoT, cloud computing and edge computing [16–18].

Key technologies for blockchain include distributed crypto ledgers, consensus mechanisms, and smart contracts. The generic blockchain architecture divides it into infrastructure, basic components, ledger, consensus, smart contract, interface, application, operation and management of systems, etc., [19]. Among them, the infrastructure layer supplies physical resources and computing drivers for the upper layer. The basic component layer provides the communication mechanism, database and cryptographic library for the blockchain network. The ledger layer is responsible for the collection of transactions and block packaging, legitimacy verification, and chaining up the block. The consensus layer coordinates and ensures the consistency of data records of all nodes in the network. The smart contract layer implements, compiles and deploys the business logic of the blockchain system in the form of code, and then completes the condition triggering and automatic execution by rules. The interface layer is mainly used to encapsulate the function modules and give a simple way to call the application layer. The management system layer is in charge of the operation and maintenance of the blockchain architecture. Finally, the blockchain services are provided for users on the application layer. As a new system delivery form, BaaS has great advantages in system scalability, ease of use, security, operation and maintenance management compared with the original deployment mode. It uses cloud computing technology for reference, adopts innovative solutions such as containers, microservices and scalable distributed cloud storage technology to provide a variety of technical options for different underlying chains, which helps to simplify the development, deployment and operation and maintenance of blockchain and improve application flexibility.

The edge computing architecture expands the edge layer on the basis of cloud computing. Therefore, the blockchain-enabled edge computing consists of the perception end, edge node, core functions of blockchain, data center and its application parts, as shown in Figure 1. Among them, the core edge nodes manage and verify their data to the chain according to the consensus mechanism through the smart contract.

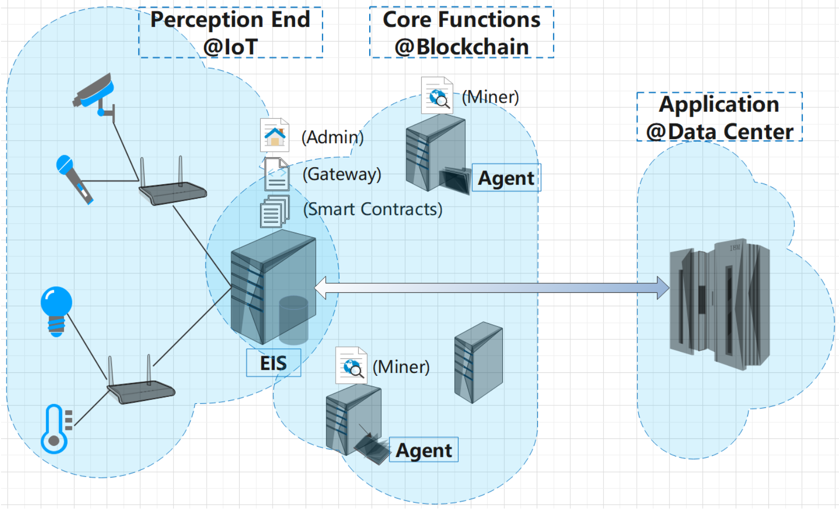


Figure 1. Diagram of blockchain-enabled edge computing.

The perception end contains all kinds of sensor terminals, which can only complete simple data processing tasks in limited resources. However, they can migrate complex computing tasks to edge nodes for processing and complete various responses by receiving processing results. Its main tasks are data acquisition and transmission, communication and response with nodes.

The core functions of a blockchain on the edge nodes: Several kinds of edge nodes are set up in order to balance efficiency and complexity. Ordinary nodes are only used as the sensor interface or connectivity to maintain IoT integrity. Special agent nodes perform special tasks, such as protocol conversion, a gateway between IoT and blockchain, and miner and query in blockchain. Manager nodes served by an Edge Intelligent Server (EIS) have strong computing and storage capabilities and integrate hardware and software to build various edge-cloud services, such as Supervisory Control And Data Acquisition (SCADA), security management, Human–Machine Interaction (HMI) for the cloud, edge application development, support for I/O and sensor compatibility and IoT protocol connectivity, intelligent data processing and management for the edge.

It benefits from EIS, where smart contracts with the consensus protocol are deployed, which can manage user credibility, device resources and node transactions. The access control, data processing, new block generation and other operations and behaviors based on blockchain technology can ensure the system security. When data need to be stored, the asymmetric cryptographic algorithm is usually used to encrypt the data to ensure that it is not leaked to other devices or users. The smart contract then records the execution of the store event as a block. Finally, a new block is added to the blockchain according to the consensus algorithm. In addition, other active events, such as access or query, are associated with the block.

The application of data center: It has the most powerful function in the whole system and is also the ultimate guarantee for realizing the blockchain-enabled edge computing task. The perceptron end migrates the tasks that cannot be undertaken to the edge node, and the edge node offloads the tasks to the data center according to the AI optimal algorithm [20]. Its application provides users with all kinds of services and is also the interface to access data on and off the chain. After a user sends out a service request, its identity is verified and then the request is sent to the smart contract on the edge layer. After the smart contract receives the user's request, it queries the user's credibility score, then judges whether the request is reasonable and whether there is malicious behavior, then decides whether to provide services and feedback to the application layer.

A good prospect for the combination of edge computing and blockchain has been demonstrated elaborately through the above-mentioned. On the one hand, it makes full use of the edge in edge computing close to the end-user to realize distributed real-time processing technology; on the other hand, it uses the decentralized consensus attribute of blockchains to further promote the scalability of distributed architecture. EIS takes on the key technical support. On the one hand, there are many services supported by EIS with computing power for edge computing and blockchain. On the other hand, it is the joint physical and logical node of the edge computing network and blockchain network. Thanks to service concepts and technologies, such as BaaS and micro-services, the different architectures are converging. In other words, the functions on the logical level can be deployed to different physical entities. Generally, it is a wise strategy to deploy key services and applications, such as admin and gateway, to a joint node.

When applying blockchain to edge computing scenarios, such as IoT and internet of vehicles, there will be a common problem as follows: There are a large number of resource constrained devices (embedded terminals) in the network, which have very limited computing, storage and processing capabilities. Therefore, how to deploy the blockchain is critical. The blockchain is deployed in the edge servers and cloud servers, and the terminal devices get the information on the blockchain through the communication connection with the edge servers [21,22]. Three types of nodes are defined by [23,24]: lightweight nodes, standard nodes, and switch-level nodes with different roles. Reference[25] introduces how a masternode is elected as a validator to manage Edgence in a decentralized way.

3. About AMS Technology and Application

Firstly, AMS, which is suitable for the IoT protocol standard, is discussed, and a typical instance is given to illustrate its applicability and practicability. Two related technologies are then outlined. One is RBAC, as the basis of AMS, to implement the basic functions of AMS, and the other is Software-defined Networking (SDN), of which the programmable software attribute can provide a multi-mode method for AMS run on the network. Finally, it is proposed to write the basic elements of RBAC into the blockchain smart contract to realize the schema of blockchain-enabled AMS for edge computing.

The underlying Access Control (AC)/Access Management (AM) process looks like this: The received packet (source IP address or source IP + source MAC) has been compared with the configured hardware address pool (AM pool), which is a list of addresses, in which the entry corresponds to a user, address information and its port. It is forwarded when found, otherwise discarded. There are four ways to implement AM: authentication, authorization, single sign-on (SSO) and federation. There are mainly two standard-based approaches related to AM in IoT. The first approach developed by the Internet Engineering Task Force (IETF) is called [26] Constrained Application Protocol (CoAP) Management Interface, and the other approach developed by Open Mobile Alliance (OMA) is called [27] Lightweight Machine to Machine (LwM2M). The AMS manages the IoT devices with CoAP in a centralized fashion via a server. Therefore, the single point of failure prevents the system from scaling. In addition, centralized architectures are designed for nodes that are static and belong to the same management community during their lifetime. In order to provide a more flexible solution, a blockchain-enabled system needs to be developed for the manager to access the resources in the edge computing. The decentralized solution has several features [28]: device nodes can be managed simultaneously by multiple managers and belong to different management domains during their lifetime in the blockchain, the managers and device nodes interact through the blockchain network, and constrained managers can easily administer device nodes with EIS.

The back-end interface of the Advantech WISE-PaaS micro-services cloud platform integrates SSO services [29]. SSO provides authentication, access control capabilities and protection mechanisms to prevent malicious attempts. Tenants can centrally manage users, assign user management rights and accessible cloud resource permissions. Its characteristics are as follows: (1) It is convenient for users to gain access of all authorized applications only with a single account, and get an improved application experience; (2) it is also convenient for administrators to maintain a set of unified user accounts only with a single account, and optimize management and reduce system vulnerabilities; (3) finally, it is convenient for developers. Advantech WISE-PaaS improves the development efficiency by integrating SSO without the additional development of an account management system. After the user login with the account password, its token will be returned. The certificate provided by the platform adopts the JSON web token (JWT) open standard (RFC 7519), which defines a simple and independent way to transfer user information as a JSON object safely with a complete Restful API. Once the user logs in, each subsequent request will bring the token in the header of request, and the back-end server will allow the user to access the paths, services and resources allowed by the token after receiving the request. There are three types of roles: super admin, admin and device admin. Super admin has the authority of all accounts to manage and create accounts for roles of admin and device admin. Admin is the administrator, which can manage and create a device admin account; that is, to create device groups, add and delete devices, power on and off of devices, device status monitoring, sensor data acquisition, and data reporting rule settings. Device admin is a general device management account, which can manage the device group assigned by the admin, including power on and off of the device, device status monitoring, sensor data acquisition, data reporting rule setting and other functions.

AMS is based on the RBAC model, which is composed of (WHO, WHAT, HOW) triple. That is, WHO performs a HOW operation on WHAT/WHICH, the core of which consists of four parts: USER, ROLE, SESSION and PERMISSION. AMS users with a key role, such as Manager/Admin, have the highest permission to manage the entire system resources and their user behavior. Agent users with gateway and miner responsibilities access edge computing and/or blockchain resources according to permission in the activity session. The normal form description of AMS will be given in the next section.

SDIoTEdge [30] (Software-Defined Internet of Things orchestration using Edge) is proposed to solve the challenge of complex IoT management. SDN architecture is a layered framework where each plane operates separately, including data, control, and application planes, which facilitates network administration at runtime and flexible network programmability. The control mechanism of SDN can reduce the edge computing architectural and implementation complexities by providing a new communication management and resource virtualization. In SDN, each OpenFlow rule has three components, including a "rule" attribute, an "action" field, and a "status", which is integrated into RBAC to form a smart contract to implement blockchain-enabled AMS.

Therefore, the schema of blockchain-enabled AMS for edge computing needs to be set up with the following principles: Given full play to EIS computing power to support the resource constrained nodes in edge computing; the OpenFlow protocol by SDN technology should be adopted to ensure the communication efficiency; use the blockchain decentralized architecture for reference to scale the management of numerous IoT devices without a single point of failure. Based on the RBAC working mode, nodes can access the information stored on both IoT and blockchain. Moreover, it makes full use of the real-time ability of the edge service close to the end-user, the communication efficiency achieved by SDN flexible network management and software programming, as well as the system scalability and data security and privacy protection brought by blockchain.

4. Model of Blockchain-Enabled Edge Computing

4.1. The Interfaces and Interactions of AMS

Centralized AMS, known as the client/server paradigm, was designed to meet the needs of traditional human–machine oriented internet scenarios where nodes are within the same trust domain, which usually requires centralized access management. However, decentralized AMS brings the following advantages: mobility, accessibility, concurrency, lightweight, scalability and transparency [31]. Consequently, the architecture does not include a zEdge Computing Node (ECN) in the blockchain and, alternatively, defines a new node called gateway that requests access control information from the blockchain on behalf of the ECN. A single smart contract defines all the operations allowed in the access control system. That contract is unique and cannot be deleted from the system. Entities called managers interact with the smart contract in order to define the access control policy of the system.

The architecture consists of the components and interfaces as following:

- Manager: It is an entity responsible for managing the access control permissions of a set of IoT devices. After registration of the IoT device under the manager's control, the managers can define specific access control permissions for them. Managers are the only entities with the ability to interact with the smart contract in order to define new policies in the AMS.
- Agent Node: It is a specific blockchain node that is the owner of the smart contract during the lifetime of the access control system to deploy the smart contract and receives an address that identifies the smart contract. The AMS is governed by the operations defined in a single smart contract. An agent is able to serve as many simultaneous requests as possible from the ECN.
- Edge Computing Networks: It is a communication network for ECN that allows constrained connectivity, which the IoT devices belong to. The current IoT communication protocols, such as CoAP support, secure channels and unique key identifiers for every device.
- Blockchain Network: Nodes can use the blockchain interface to store and globally access the specific devices by an access control policy. The information is fully decentralized and tamper-proof. All the operations are defined in the smart contract. Once an operation is triggered through a transaction, the miners will keep the information of the transaction globally accessible. Querying information from the blockchain does not incur any fee.
- Gateway: It is an interface that translates the information encoded in CoAP messages by the ECN into JSON-RPC messages understandable by the blockchain nodes, which is connected directly with a blockchain node.

The entities will be identified by their public keys with their names, and the permissions of policies that allow or prevent others from viewing, modifying or executing them can be defined through the AccessControl operations with the data structure in a smart contract, as shown in Table 1.

Table 1. AccessControl operations in smart contract.

No.	Name	Operations	Note
1	Init	__init__	Initialize the system and import the access policy into smart contract.
2	RegNode	Register(Type = 1, 'AttMan')	Register a device as a node.
3	RegANode	Register(Type = 2, '')	Register a device as an agent node.
4	RegMan	Register(Type = 3, '')	Register a device as a manager.
5	AddUser	U∪{u}	Add user after login.
6	AddRole	R∪{r}	Add the role into set R.
7	AssiUR	U × R	Assign users to roles.
8	BindAU	ACT × R	Bind users to activities.
9	Query	Query(Nodes, Users, Roles)	Query access policy in detail.

According to the defined operation, the behavior interacts in the following order:

1. Setting up the management blockchain network. AMS is created in the blockchain network, in which the agent node deploys the smart contract, which defines all the operations of AMS and where the address is used to identify.

2. Registering nodes of the managers. They are used to manage the access control policies and the IoT devices that belong to their manager. Any blockchain node in the AMS can be registered as a manager. Once the address of the smart contract is obtained, it can register itself sending a transaction to the function RegisterManager defined in the smart contract.
3. Defining the policy for those aforementioned components. The policy is a set of constraints and rules, which are used to describe and regulate system state and activity. Managers can define access control rules for the resources of their IoT devices, and enforce the policy creating a transaction towards the smart contract.
4. Discovering the policy. An IoT device first needs to discover the gateway's IP address, which then translates the device's message into an RPC message and sends it to the agent in the blockchain network attached to it. The operation queries the information from the blockchain.

All the managers in the architecture are externally owned accounts while the smart contract is deployed under a contract account. The operations query the information of the blockchain, which are called Call, and are used in the gateways to invoke a function in a smart contract. In general, the gateway is a JavaScript interface that helps the IoT devices to connect with the blockchain network, and uses the JavaScript API to communicate with the nodes through RPC calls and a CoAP JavaScript library. Accordingly, device nodes are implemented using the LibCoAP library, which can support transport layer security to automatically generate a public/private key per device in the tinydtls framework.

4.2. The Architecture and Behavior with the CPN Model

The AMS core model based on RBAC contains a basic element entity set as follows: user set U, role set R, permission set P, session set S [32]. The corresponding role set, operation set and activity set are expressed as: R = {Man, Nod, ANod, Gate}, OP = { AddUser,AddRole,BindUA,BindAR }, ACT = (aName,aType,aIn,aOut), where R and OP are associated with the entity of architecture and operations of the smart contract in the previous section, respectively. In ACT, aName represents the name or ID of the activity, aType represents the type of the activity, aIn is the input set of the activity, and aOut is the output set of the activity. The activities can be divided into the following seven types of dependencies ∀ a,b∈ ACT: sequence relationship as (a,b); parallel relationship as (a;b); conditional branch as (Cond ?a,b); (a AND b); (a OR b); (a XOR b); loop relationship as (Loop a).

The association between the above entities and object operations are described as follows:

- The assignment relationships are represented by '×'. For example, the assignment relationship between a user and a role was represented as UA ⊆ U×R, and the assignment relationship between permissions and a role was represented as PA ⊆ P×R.
- The mapping relationships are represented by a '→'. The mapping relationship between the session and user is represented by 'user: S →U', and the mapping relationship between session and role is represented by 'R : S→R'. Where R(s) ⊆{ r | (user(s),r∈UA } and | R | = 2^R, and S has a permission set of $\bigcup_{r \in R(s)}$ {p | (p,r ∈PA}.
- The binding operations in the activity are described as follows:
  AddUser: If u∉U then U = U∪{u};
  AddRole: If r∉R then R = R∪{r};
  BindUA: If u∈U and a∈ACT, ¬(u∈U, assigned_user(u) = a) then assigned_user(u) = a;
  BindAR: If a∈ACT and ∃ r∈R, then bound_user(a) = r.

Furthermore, a Petri net is introduced to build an RBAC model of the architecture and behavior, which is a directed net composed of four parts: place, transition, token and directed arc, where the place describes the state of the system and transitions describe the events of the system. A Petri net is represented as a quintuple PN = (P,T, I, O, M). The dynamic process of an event can be described by the transition identified with the state. It has been widely used in system analysis and modeling characterized by asynchrony, concurrency, distribution and uncertainty, and can capture system state and events simultaneously. Therefore, it is very suitable to use a Petri net to analyze security policy based on state, and it is beneficial to deduce the execution model based on an event in the system. The correctness of the policy is verified to ensure its security.

A Colored Petri Net (CPN) is an extension of a Petri net. Combining the graphical interface representation of a Petri net with the data structure of the CPN ML programming language, it can be widely used in collaborative design and process control-analysis in workflow management, which is not only convenient for the intuitive description of the system structure model, but also for computer simulation and formal verification of the model [33]. CPN is represented as a six-tuple (P,T, C, I, O, M), where C is the color set. The color set of place P_i is represented as $C(p_i) = \{a_{i,1}, \ldots, a_{i,u_i}\}$, $u_i = |C(p_i)|$, i=1,...,n, and the color set of transition T_j as $C(t_j) = \{b_{j,1}, \ldots, b_{j,v_j}\}$, $v_j = |C(t_j)|$, j=1,...,m. Colors are defined as a set of variable types to specify the token type, arc variable type, and various functions. This corresponds to defining the data structure of PN and declaring the color type, function, operation and variable of CPN ML.

The typical colors [34] in the CPN-based RBAC policy model are described as follows:

- colset cR = list R: A list of colors used to represent role tokens;
- colset cUR = product U * R : A tuple (u_i, r_i) used to represent the token color that defines the assignment relationship of user and role;
- colset cSR = product S * R : Used to represent that the set of roles activated in the session is determined by the cR. For example, (s_1, (r_1, r_2, r_3)) indicates that the roles to be activated by the user in session s1 are r_1, r_2, and r_3;
- colset cURS = product U * R * S timed : The triple-tuples (u, r, s) indicates that user u activates role r in session s;
- colset cCmd = with assign | deassign | enable | disable | active | deactive : It is used to define six kinds of events in the model, namely: user-role assignment, user-role deassign, role validation, role invalidation, user-role activation, and user-role deactivation;
- colset cEvt = product Cmd * U * R *S : The quad-tuples (cmd, u, r, s) define the expression of events.

The AMS model based on RBAC is built by CPN, and the static structure and dynamic behavior of the privilege management mechanism are described and analyzed. The running rules of the model are based on the dynamic role authorization, so as to associate users, operations and activities, and provide flexible and secure access policies for various cooperative objects.

5. Simulation of the CPN Model

CPN-Tools provides both graphical and formal expressions to integrate simulation, representation, analysis and verification, which can express concurrency, conflict and causal dependency. It supports Meta Language and incorporates powerful model validation, such as state space analysis, timing simulation, and functional analysis, of which the blocks could be customized to extract data for user performance evaluation.

5.1. RBAC Operations

The core of the RBAC model is the effective and secure access to resources based on the policies formed by users, roles, permissions and their constraint relationships. First, the relevant CPN model is set up based on the typical operation for roles, which will not be repeated in detail about other factors. The corresponding attributes are shown in Table 2.

Taking the above operations as a sub-module, the AMS behavior with a CPN model can be built according to the business activities and workflow in a blockchain-enabled edge computing scenario.

Table 2. Role attributes in the CPN model.

	Role Assignment	Role Activation	Role Binding
Initial state	assigned user token in U; assigned role token in R; the current assignment relationship token between users and roles in UR; inheritance relationships between roles in RR.	current effective role token in Reff; currently allowed activated role token in Ract; user token that currently allowed activated role in URact.	assigned user token in U; assigned role token in R; assigned user's activities token in ACT.
Places	place U, place R, place UR, place RR.	place Reff, place Ract, place URact, place UR.	place U, place R, place ACT.
Transition	Transition assign: assert the assign command according to the assigned user token in U and the assigned role token in R; Transition deassign: assert the deassign command according to the assigned user-role token in UR.	Transition active: assert the active command according to the above initial tokens in Reff, Ract and URact; Transition deactive: assert the deactive command according to the user token that currently allowed activated role in URact.	Transition binding: assert the binding command according to the above initial tokens in U, R and ACT.

5.2. Behavior of AMS

From the architecture view, blockchain-enabled AMS has both domains of edge computing and blockchain, which are connected through agent interfaces, and the behavior includes elements of operations, activities and role interactions. The typical process with the CPN model [35] is shown in Figure 2.

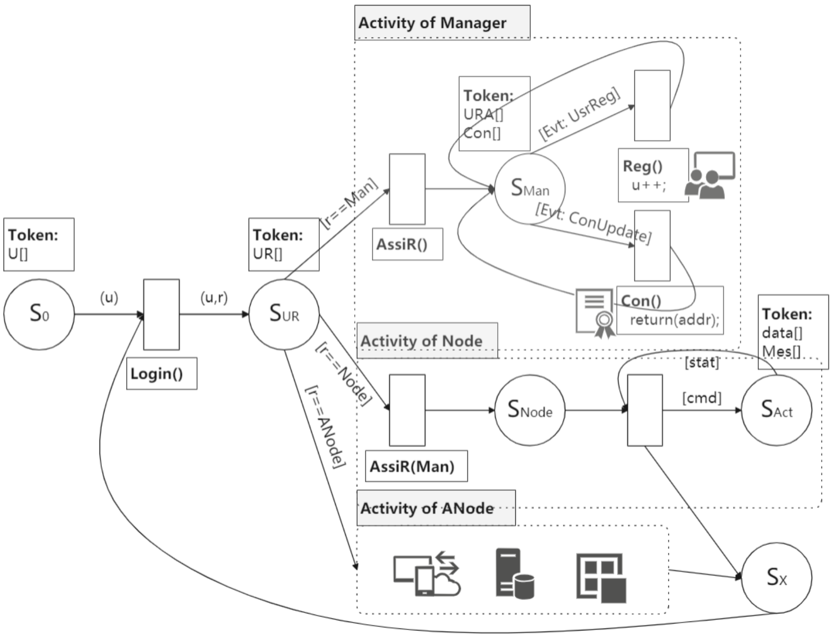


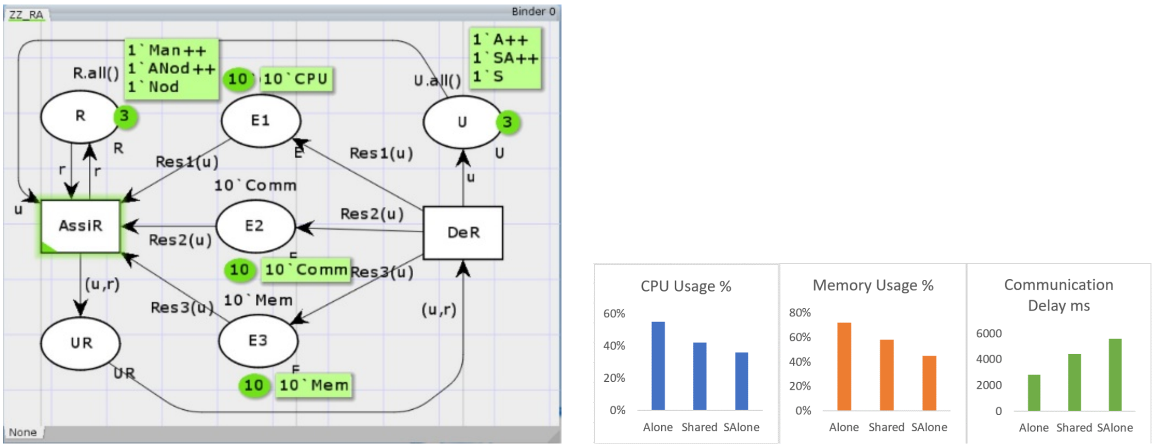
Figure 2. CPN model of blockchain-enabled edge computing.

There are three main activities in Figure 3, namely Activity of Management (AM), Activity of Node (AN), and Activity of AgentNode (AAN), where AAN contains three basic services: Service of Blockchain Miner (SBM), Service of IoT (SIoT), and Service of Cloud (SCloud). After logging in, users will participate in the following activities according to their respective roles:

1. AM. Manager is mainly responsible for managing users and role authorization, as well as providing smart contract services. When user Node and user AgentNode log in, the manager adds them to the current UR list, which can be applied in activities. After

the smart contract is updated or a new user registers, it will broadcast the contract address to provide blockchain services.

2. AN: Activities of a normal node user are relatively simple. The given task is completed according to the activity requirements after the role permission, and the corresponding state is returned. The specific operations are "receive/send the data/message" for the task.
3. AAN: Considering the real-time requirements of the IOT network, the representative consensus mechanism of the blockchain is adopted, and user AgentNodes are selected to perform the miner task, which record the RBAC operation transactions in the smart contract in the blockchain. To connect the IoT and blockchain networks, the user AgentNodes are responsible for gateway communications and data processing over their protocols. The data transmission between the edge and cloud is also a responsibility of user AgentNodes. Restful protocol is a good means to realize the request/response.



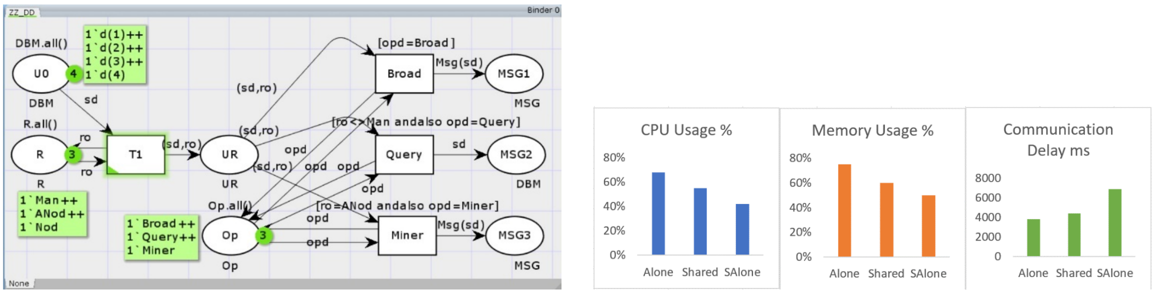
(**a**) CPN model (**b**) Operational Performance

Figure 3. Resource allocation.

After being released by the Manager, user Nodes and AgentNodes can re-login and be assigned a role to perform different activities, so as to effectively utilize resources and achieve better benefits of the system. When implementing blockchain-enabled AMS, critical services (such as user management) and special services (such as SBM) are deployed on a high-performance EIS, and user Nodes simply run on embedded terminals.

5.3. Performance Analysis

Resource allocation and distributed data access are important activity scenarios in blockchain-enabled edge computing. The former reflects system efficiency, while the latter imitates the operation of a distributed ledger in a blockchain. The simulation results of experiments on operational performance and accessibility under different deployments are shown in Figures 3 and 4.



(**a**) CPN model (**b**) Operational Performance

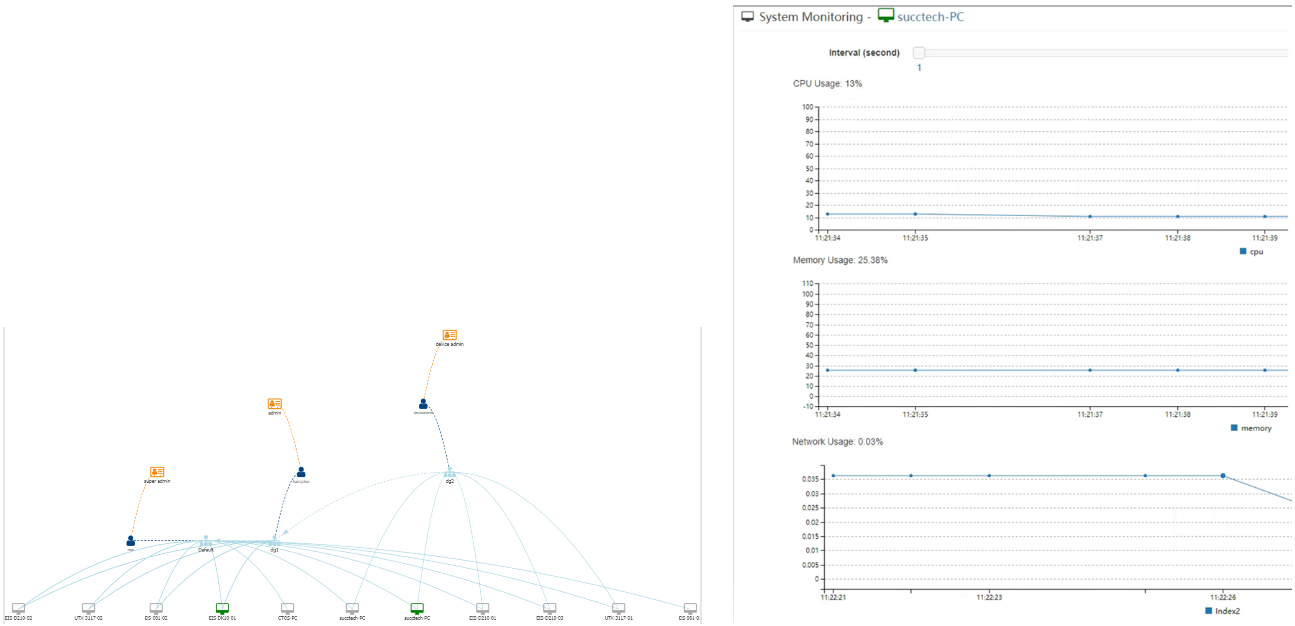
Figure 4. Distributed data access.

There are three typical deployment cases: AloneNode (where all users are on one node), StandaloneNode (where each user is on a different node) and SharedNode (where users in the same role share nodes). ICT resource allocation varies in different cases, as shown in place E1 (CPU resource), place E2 (communication resource) and place E3 (memory resource) in Figure 3a. The resources are allocated according to the operation and withdrawn after out of the activity. In the distributed data access case, the initiator node sends messages to the other nodes. The distributed message communication among different user operations has been implemented by "fun MSG(s)= DD.mult(1's, DBM.all () – 1's)" in Figure 4a. The state information of edge computing has been aggregated by components, such as Ceph, K8S, RabbitMQ and Prometheus, and finally visually displayed by Grafana graphical tools. Accordingly, the operational performance of the corresponding CPN model is presented in Figures 3b and 4b. Although the communication delay of case AloneNode is the smallest, the resource consumption of this mode will greatly reduce the processing speed. DSE is carried out to trade-off consumption and performance in order to obtain the optimal system with CPN behavior validation and platform experimental statistics.

When there are multiple transitions satisfying the occurrence condition, the simulation will select one of them to occur. The state space generating tree in the CPN model is used to get the reachable states of system, as well as the relationship between the states. For example, if the activity–role binding relationship is reachable, at least one node and transition sequence in the existing state space can satisfy the following conditions: $(M_i(Pr) > 1) \wedge (M_i(P_{rs}) > 1) \wedge (M_0[\sigma > M_i])$, where M is the initial token. It can be verified that the states in blockchain-enabled edge computing are reachable with no deadlock by activity in the typical path and result set in the state space of CPN-tools.

The performance and reachability of AMS for blockchain-enabled edge computing are proved by the above simulation results. Furthermore, based on the instance of the Advantech WISE-PaaS micro-services cloud platform with the SSO mode described in Section 3, its task statistics of an AMS for database access are shown in Figure 5:

Figure 5a shows the topological relationship among three users of the system: root, Huangchao and deviceadmin, whose accounts are root, Huangchao and deviceadmin, corresponding to three roles of super admin, admin and device admin, respectively. The devices are divided into two groups: DG1 and DG2. The Huangchao account is used to manage all devices under DG1, the DeviceAdmin account is used to manage all devices under DG2, and the root account can manage all devices. Figure 5b shows the CPU usage, memory usage and network usage of EIS.



(**a**) RBAC on WISE-PaaS Platform (**b**) Status Statistics on WISE-PaaS Platform

Figure 5. Status statistics of PG database on WISE-PaaS platform.

6. Conclusions

The real-time processing on edge can be provided by edge computing, and the security and privacy through the key information stored on the blockchain can be ensured. Based on this, the AMS model of blockchain enabled edge computing is proposed and verified by CPN tools. Important information such as user identity, access control policy and index data can be stored on the chain to confirm the behavior in the access process in a more reliable and secure way. The formal model is set up by CPN to analyze the AMS policy mechanism based on dynamic role permission, operation and activity. State space analysis shows that the model can verify the constraints, such as activity dependency, reachability of user role binding and deadlock avoidance, and meet the various access control requirements in blockchain-enabled edge computing. The goal was to provide a generic, easy-to-manage and decentralized AMS.

Meanwhile, the above AMS policy mechanism is introduced into the WISE-PaaS platform to further verify its practical application. Performance gain can be obtained by DSE and optimization according to the quantitative performance statistics of experimental data deployed in WISE-PaaS.

There are still two issues that are worth further research:

1. With the support of ICT resources provided by EIS in edge, the real-time performance of blockchain-enabled edge computing will be further improved and optimized.
2. The fine-grained access control mode should be explored to realize flexible and intelligent AMS based on smart contract.

Because of its key features of decentralization, distribution and security, blockchain-enabled edge computing is believed to be able to effectively deal with scalability and security privacy issues of a system. It will become one of the information infrastructures in the future society, and realize integration innovation with cloud computing, big data, IoT and other information technologies.

Author Contributions: Correspondence author Y.Z. uses description-simulation methodology to build AMS CPN model, and uses CPN tools for simulation and verification. C.H. built WISE-PaaS platform to collect test data. Z.H.'s main work is to manage the manuscript and visualization. A.A.-D. and M.A.-D. provide application scenarios for the paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the JIT fund grant number 40620025, the fund of State Key Laboratory of Computer Architecture (ICT, CAS) grant number CARCHA202010, the fund of Science and Technology Development Center, Ministry of Education grant number 2018A03040, and Deanship of Scientific Research at King Saud University for funding this work through research group grant number RGP-264.

Data Availability Statement: The model and data used to support the results of this study are available from the corresponding author upon request.

Acknowledgments: The authors acknowledge the JIT fund (Grant No.40620025), the fund of State Key Laboratory of Computer Architecture (ICT, CAS) (Grant No.CARCHA202010), the fund of Science and Technology Development Center,Ministry of Education (Grant No. 2018A03040), Deanship of Scientific Research at King Saud University for funding this work through research group (No. RGP-264).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, X.; Wang, Y.; Lu, S.; Liu, L.; Xu, L.; Shi, W. OpenEI: An Open Framework for Edge Intelligence. In Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS), Dallas, TX, USA, 7–10 July 2019; pp. 1840–1851.
2. Zhou, Z.; Chen, X.; Li, E.; Zeng, L.; Luo, K.; Zhang, J. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proc. IEEE* **2019**, *107*, 1738–1762. [CrossRef]
3. Deng, S.; Zhao, H.; Fang, W.; Yin, J.; Dustdar, S.; Zomaya, A.Y. Edge intelligence: The confluence of edge computing and artificial intelligence. *IEEE Internet Things J.* **2020**, *99*. . [CrossRef]
4. Singh, S.; Hosen, A.S.M.S.; Yoon, B. Blockchain Security Attacks, Challenges, and Solutions for the Future Distributed IoT Network. *IEEE Access* **2021**, *9*, 13938–13959.
5. Singh, S.; Sharma, P.K.; Yoon, B.; Shojafar, M.; Cho, G. H.; Ra, I.-H. Convergence of blockchain and artificial intelligence in IoT network for the sustainable smart city. *Sustain. Cities Soc.* **2020**, *63*, 102364. [CrossRef]
6. Li, T.; Ren, Y.; Xia, J. Blockchain Queuing Model with Non-Preemptive Limited-Priority. *Intell. Autom. Soft Comput.* **2020**, *26*, 1111–1122. [CrossRef]
7. Bordel, B.; Alcarria, R.; Martin, D.; Sanchez-Picot, A. Trust provision in the internet of things using transversal blockchain networks. *Intell. Autom. Soft Comput.* **2019**, *25*, 155–170. [CrossRef]
8. Wang, J.; Chen, W.; Wang, L.; Ren, Y.; Sherratt, R.S. Blockchain-based data storage mechanism for industrial Internet of things. *Intell. Autom. Soft Comput.* **2020**, *26*, 1157–1172. [CrossRef]
9. Yang, R.; Yu, F.R.; Si, P.; Yang, Z.; Zhang, Y. Integrated blockchain and edge computing systems: A survey, some research issues and challenges. *IEEE Commun. Surv. Tutorials* **2019**, *21*, 1508–1532. [CrossRef]
10. Fan, K.; Pan, Q.; Zhang, K.; Bai, Y.; Sun, S.; Li, H.; Yang, Y. A Secure and Verifiable Data Sharing Scheme Based on Blockchain in Vehicular Social Networks. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5826–5835. [CrossRef]
11. Koulu, R. Blockchains and online dispute resolution: Smart contracts as an alternative to enforcement. *SCRIPTed* **2016**, *13*, 40. [CrossRef]
12. IBM. IBM Trusted Identity[EB/OL]. Available online: https://www.ibm.com/blockchain/solutions/identity (accessed on 1 March 2020).
13. Huckle, S.; Bhattacharya, R.; White, M.; Beloff, N. Internet of things, blockchain and shared economy applications. *Procedia Comput. Sci.* **2016**, *98*, 461–466. [CrossRef]
14. Hurich, P. The virtual is real: An argument for characterizing bitcoins as private property. *Bank. Financ. Law Rev.* **2016**, *31*, 573.
15. Dorri, A.; Kanhere, S.S.; Jurdak, R.; Gauravaraml, P. Blockchain for IoT security and privacy: The case study of a smart home. In Proceedings of the 2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops), Kailua-Kona, HI, USA, 13 March 2017; pp. 618–623.
16. Zhang, Y.; Wen, J. The IoT electric business model: Using blockchain technology for the internet of things. *Peer-to-Peer Netw. Appl.* **2017**, *10*, 983–994. [CrossRef]
17. Stanciu A. Blockchain based distributed control system for edge computing. In Proceedings of the 2017 21st International Conference on Control Systems and Computer Science (CSCS), Bucharest, Romania, 29–31 May 2017; pp. 667–671.
18. Ouaddah, A.; Abou, Elkalam, A.; Ait, Ouahman, A. FairAccess: A new Blockchain-based access control framework for the Internet of Things. *Secur. Commun. Netw.* **2016**, *9*, 5943–5964. [CrossRef]
19. China Academy of Information and Communications Technology. White Paper in Blockchain. 2019. Available online: http://www.caict.ac.cn/kxyj/qwfb/bps/201911/P020191108365460712077.pdf (accessed on 20 April 2021.)
20. Dai, Y. Edge computing-based tasks offloading and block caching for mobile blockchain. *Comput. Mater. Contin.* **2020**, *62*, 905–915.
21. Sharma, P.K.; Chen, M.Y.; Park, J.H. A software defined fog node based distributed blockchain cloud architecture for IoT. *IEEE Access* **2017**, *6*, 115–124. [CrossRef]

22. Li, C.; Zhang, L.J. A Blockchain Based New Secure Multi-Layer Network Model for Internet of Things. In Proceedings of the 2017 IEEE International Congress on Internet of Things (ICIOT), Honolulu, HI, USA, 25–30 June 2017; pp. 33–41.
23. Veena, P.; Panikkar, S.; Nair, S.; Brody, P. Empowering the edge-practical insights on a decentralized internet of things. *IBM Inst. Bus. Value* **2015**, *17*. Available online: https://www.ibm.com/downloads/cas/2NZLY7XJ (accessed on 20 April 2021).
24. Panikkar, S.; Nair, S.; Brody, P.; Pureswaran, V. Adept: An iot practitioner perspective. Available online: https://www.windley.com/archives/2015/02/ibms_adept_project_rebooting_the_internet_of_things.shtml (accessed on 20 April 2021).
25. Xu, J.; Wang, S.; Zhou, A.; Yang, F. Edgence: A blockchain-enabled edge-computing platform for intelligent IoT-based dApps. *China Commun.* **2020**, *17*, 78–87. [CrossRef]
26. The Constrained Application Protocol (CoAP). Available online: https://www.rfc-editor.org/pdfrfc/rfc7252.txt.pdf (accessed on 20 April 2021.)
27. LwM2M v1.1. Available online: http://www.openmobilealliance.org/release/LightweightM2M/Lightweight_Machine_to_Machine-v1_1-OMASpecworks.pdf (accessed on 20 April 2021).
28. Novo, O. Scalable access management in IoT using blockchain: A performance evaluation. *IEEE Internet Things J.* **2018**, *6*, 4694–4701. [CrossRef]
29. ADVANTECH. SSO Service of WISE-PaaS Cloud Platform. 2018. Available online: https://docs.wise-paas.advantech.com.cn/en/Guides_and_API_References/Cloud_Services/SSO/1581403317441085734/v1.0.2 (accessed on 20 April 2021)
30. Rafique, W.; Qi, L.; Yaqoob, I.; Imran, M.; Rasool, R.U.; Dou, W. Complementing IoT services through software defined networking and edge computing: A comprehensive survey. *IEEE Commun. Surv. Tutorials* **2020**, *22*, 1761–1804. [CrossRef]
31. Novo O. Blockchain meets IoT: An architecture for scalable access management in IoT. *IEEE Internet Things J.* **2018**, *5*, 1184–1195. [CrossRef]
32. Zhao, G. Cpn-Based Specification and Verification for Rbac with Temporal Constraints. Ph.D. Thesis, Harbin Institute of Technology, Harbin, China, 2010.
33. An Y.S.; Luo, B.J.; Zhao, X.M.; Li, R.H. Modeling and Verification of Dynamic Management of Cooperative Permission Based on Colored Petri Nets. *J. Chin. Comput. Syst.* **2012**, *33*, 1972–1977.
34. Bao, N. Specification and Conflict Detection for Gtrbac in Multi-Domain Environment. Ph.D. Thesis, Harbin Institute of Technology, Harbin, China, 2013.
35. Zhai, Z.; Xi, J.; Lu, Y.; Guo, Y. An Access Control Model with Task-State Sensitivity and Its CPN Simulation. *J. Xi'An Jiaotong Univ.* **2012**, *12*, 85–91.