Article

Random Authentication Node Selection Mechanism in Block Network for Meta-Mobility Service Data Reliability

Jinsu Kim , Eunsun Choi , Byung-Gyu Kim , Namje Park *

Posted Date: 17 August 2023

doi: 10.20944/preprints202308.1257.v1

Keywords: Blockchain; Mobility; Random Selection; Encoding; Token

Article

Random Authentication Node Selection Mechanism in Block Network for Meta-Mobility Service Data Reliability

Jinsu Kim [1], Eunsun Choi [2], Byung-Gyu Kim [3] and Namje Park [1,2,4,*]

[1] Department of Convergence Information Security, Graduate School, Jeju National University, Jeju City 63243, Republic of Korea; kimjinsu@jejunu.ac.kr
[2] Creative Education Base Center, Jeju National University, 63294, Republic of Korea
[3] Dept. of Artificial Intelligence Engineering, Sookmyung Women's University, Seoul City 04310, Republic of Korea
[4] Department of Computer Education, Teachers College, Jeju National University, Jeju City 63243, Republic of Korea
* Correspondence: namjepark@jejunu.ac.kr

Received: 31 March 2020; Accepted: 27 May 2020; Published: date

Abstract. Various elements such as evolutions in IoT services resulting from sensoring by vehicle parts and advances in small communication technology devices have significantly impacted the mass spread of mobility services that are provided to users in need of limited resources. In particular, business models are progressing away from one-off cost towards longer term cost as represented by shared services utilizing kick-boards or bicycles and subscription services for vehicle software. Advances in shared mobility services as described are calling for solutions that can enhance reliability of data aggregate by users leveraging mobility services in the next-generation mobility areas. However, mining process to renew status, ensures continued network communication and block creation demands high performance in public block chain. This thesis proposes random certificate node selection mechanism in block network that creates blocks via node that has tokens issued for block creation and lets only specific nodes selected by encrypting token information acquires token. In the proposed mechanism, all clients belonging to the block network can have the authority to create blocks, and block creation can be performed only through an authentication means called a token, preventing indiscriminate block creation. In addition, centralization of block generation is prevented by allowing clients designated by the token group to create the next block to create the token group.

Keywords: Blockchain; Mobility; Random Selection; Encoding; Token

1. Introduction

Unlike internal combustion engine vehicles, today's vehicles are getting more electric and this means increasing electrification of various devices that are controlling driving by subjecting many vehicles parts to control by sensors [1,2]. On top of this, advances in IoT communication technology helps personally-owned bicycles and kick-boards be remote controlled and protected against theft, which in turn, is an accelerator for shared services. The mobility sector is currently focusing on building shared mobility while also conducting research in the area of meta-mobility, which combines virtual environments of metaverses with mobile services, as the next generation of mobility. Meta-mobility aims to provide users with immersive experiences through various mobile services. In simple terms, research is being conducted on meta-mobility, which combines various mobile services and immersive experiences in the field, while emphasizing the establishment of shared mobility. The mobility services are being offered as subscription-based services, and the business models are evolving to generate sustainable and stable revenue by integrating metaverses [3–5].

At present, subscription services are limited only to such functions as autonomous driving and smart phone vehicle control system to improve user convenience. As mobility services get more intelligent there is greater need to protect numerous data created along the way. At the same time, electric vehicle control as opposed to the analogous vehicle control that is independent raises the risk of a vehicle owner hijacked of driving power by a third party infiltrating from a foreign network, which could result in an accident [6–8]. Therefore, future mobility services require means to securely protect data in order to not only provide simple network communication but also deliver stable virtual environments to users [9].

Block chain is apparently an attractive solution in that it guarantees integrity of mobility data and helps contracts to be implemented on a real time basis. Yet, there are many challenges in applying it in a limited mobility environment. It guarantees transparent transactions and reinforces integrity of block data as blocks increase, which makes it a good choice in many areas [10–12]. In particular, PoW, which does not restrict participants, faces more issues to be addressed in order to acquire compensation for creating transaction blocks. Mobility environment, in general, does not require high performance and shared mobility such as bicycles and kick-boards should consider low performance and energy efficiency since it is based on IoT environment. This requires PoW to perform computation in order to continuously get hash value to compete for block creation. This also suggests limitations in applying in shared mobility environment in that client-to-client communication has to be made [13–16].

Compared to PoW, which exhausts resources for hash computation to create blocks, PoS that acquires block creation authority corresponding to one’s share can substitute resources consumed for hash computation by proving its own share. This enables application with lower performance level. However, the fact it acquires authority as per share and subsequent compensation means it is prone to fixation [17,18].

Mechanism proposed in this thesis provides tokens to create blocks by selecting random nodes to reinforce data integrity in a limited scope whereby share from block creation does not affect mobility service in itself. Also, the use of encrypted tokens in the course of sending them decodes encrypted tokens only in specific nodes ensures communication without disclosing selected nodes in network.

2. Related Studies

2.1. Shared Mobility Service Network

There are many network standards that include NB-IoT(NarrowBand-Internet of Things), LoRa(Long Range) and eMTC(enhanced Machine-Type Communication) for application to mobility services. In particular, low power, low performance and higher energy efficiency are in demand for use in such limited performance as IoT. IoT network standard focuses more on energy efficiency and data transmission distance than on speed itself since it has to be applied in the limited IoT environment [19,20].

LoRa is a wireless communication technology for low power, long-distance communication. It derives from CSS(Chirp Spead Spectrum) technology. Its low transmission means it is applicable to IoT that sends small-sized data and transmission speed is 290bps in 14km coverage and 5470bps in 2km coverage. It consumes less battery since it transmits data and maintains slip with consistent time interval [21].

NB-IoT is the standard low power, broad communication technology established by 3GPP(3rdGenerationPartnershipProject) mobile communication standardization organization. With a narrow bandwidth of 180kHz, it supports 250kbps in data transmission speed and broad service beyond10km. It is suitable for fixed smart service since it does not support mobility. It is applicable to an IoT environment characterized by less frequent data use, low power consumption and low mobility. It is also ideal for metering sensors like water, gas, electricity, temperature as well as smart lights, location tracing devices that are located in remote areas [22].

eMTC, which is also known as LTE cat M1, was standardized by 3GPP's Release 13. It provides up to 1Mbps in data transmission speed at 1.4MHz bandwidth. It also provides VoLTE(Voice over Long Term Evolution). Its high data transmission speed makes it a good application for technologies that require heavy data such as tracking mobile objects or real time services [23].

As shown in [Table 1], which compares IoT communication technologies, environment with less mobility does not require real-time processing from LoRa or NB-IoT technologies. Rather, they are applied to environments with smaller data transmission volume. In mobility services, on the other hand, data has to be processed real time and mobility has to be guaranteed, too. In mobility services that require mobility and real-time processing such as kick-boards in Korea, eMTC is applied.

Table 1. Comparison of IoT networks.

Category	LoRa	NB-IoT	eMTC
Coverage	~10km	~15km	~11km
Bandwidth	920-925MHz(korea)	180kHz	1.4MHz
Transmission speed	10kbps	~250kbps	~1Mbps
Battery life	~10 years	~10 years	~10 years

2.2. Blockchain Consensus Mechanism

Block chain consensus mechanisms include PoW(Proof of Work), PoS(Proof of Stake) and DPoS(Delegated Proof of Stake), etc.

PoW, which is the most well-known mechanism, gives block creation authority to the node that used the largest amount of resources to creating block chain's block. The process of substituting nonce value on a repeated basis to find hash value below target value, thereby solve largest number of problems and have the authority to create blocks is called mining [24–26]. Bit coin, which is the most popular example of PoW, is huge in size and its network is protected from attack by 51% capable of exercising block's forgery by participating node 51%, which is one of the weaknesses of block chain. This is because the economic value required to acquire node's 51% computing power to prevent data forgery is exponentially high and therefore poor in efficiency [27]. Difficulty of mining is set in a way to produce a certain interval as per computation level of nodes taking part in the mining process, which subsequently, aggravated competition for mining using high-specification equipment. This, in turn, made mining more challenging and mining equipment that is higher in performance in need. Growing necessity of high-performance computing equipment is consuming more electric power and causing too much waste of energy [28,29].

PoS is an algorithm that grants decision-making authority to node based on shares. It takes parts in creating blocks by proving its shares to the block and therefore does not demand resources consumed in the course of mining, which is unlike PoW that competes to acquire compensation [30,31]. As such, it consumes less energy and all nodes that has shares take part in decision-making since block's updating authority is matched by shares. One key example operating in PoS is ethereum. Its consensus mechanism was based on PoW at first but has since converted to PoS. Thanks to this conversion, it not only consumes less electricity in mining but also mitigates environmental issues such as heat generated during mining. These changes are not without downsides. Compensation in PoS method makes distribution based on shares. This means higher compensation to nodes with higher shares, which subsequently has bigger impact on the network and issues associated with coins concentrated in specific nodes [32,33].

DPoS method is an algorithm whereby nodes can exercise their voting rights as per shares and engages in decision-making via the selected proxy. Since it appoints a proxy that will make decisions on behalf by acquiring voting rights per share transaction approval by a few representatives can accelerate processing speed relative to PoS solution where all nodes take part [34]. However, that blocks are created only by a few selected proxies is a far departure from meeting the objective of block chain, which is decentralization. Another downside is that participants with less shares are selected as proxy. Those with many shares in DPoS environment do not want to see their shares undermined and this can promote participants' reliability. On the contrary, such participants are at higher risk of undermining shares and this could adversely impact the block chain environment [35,36].

2.3. Related Research Trend

Various studies on mobility services leveraging block chain are in progress in sync with advances in shared mobility services. In particular, there are many studies underway to share vehicles and to apply in low-performance environment without limiting the scope of research to apply block chain in mobility environments based on IoT.

In one research, Sophia examined shared mobility environment based on IoT and particularly focused on cases applying to vehicle sharing and vehicle lease environment. The research proposed architecture of a platform based on IoT and block chain to promote shared mobility by combining vehicle sharing and lease and realized a platform that simplifies sharing and lease procedures. In a research by Madhusudan [37], it was mentioned that intelligent vehicles are recording innovative growth but there are a lot of security vulnerabilities at the same time and challenges in safely sharing data with traditional way of security protection. As a solution, the research suggested determining security elements that are required for data sharing and sharing intelligent vehicle data. In another one by Madhusudan [38], he mentioned that intelligent vehicles perform vehicle-to-object communication based on the Internet and such communication environment brings with it a wide variety of security vulnerabilities. Major issues witnessed in intelligent vehicle communication include reception in communication channels, data reliability, accuracy and security, and studies to build reliable intelligent transportation system by applying block chain as solutions were carried out.

Researches on shared mobility environment are not limited to vehicles. In fact, researches are being made in various areas like bicycle sharing services. Hanyue set leakage of users' personal information by shared bicycles and property damage as key issues in his research [39]. As a solution to this, he proposed bicycle sharing system based on block chain service platform and C2C(consumer-to-consumer) shared operation method to address limitations of the existing centralized method. Daozhi's research [40] looked into a system that taps into smart contract to prevent issues arising from companies declining to return user deposit when they discontinue bicycle sharing service in the course of the rapid growth of the platform.

3. Proposal of Block Network Random Authentication Node Selection Mechanism

Objective of this thesis is to record information such as user information, mobility device information and payment information in shared mobility environment in block networks. General mobility environment is defined by equipment put to use for services by companies. Hence, block networks in mobility environment do not need to register countless number of users in the open network. Block networks, therefore, form a private block chain structure accessible only by certified users. Private block chains, however, can be modulated by a few upper nodes dominated in the process of forming transactions since authority over user participation and block creation is handled by an upper body. Hence, upper nodes go through authentication process to access block networks in this thesis and block creation authority is performed by nodes in the block network. Block creation authority in the block network is performed by tokens, which are authentication means to acquire authority to create blocks. As a way to transmit random nodes on the block network after blocks are creation it averts monopoly.

[Figure 1] is the general concept of the mechanism proposed. Mobility device is a concept covering kick-boards, bicycles and intelligent vehicles that are part of the shared network. Devices are managed by shared service provider's server and only registered devices are joined as members of the block network. Each device becomes a node in the network. Number of tokens remains consistent as per consensus by groups that provide the service. The initial token is issued by service provider groups and tokens are transmitted after selecting random nodes in the block network. Node aggregates that received the token build consensus network to create blocks among nodes that have tokens and create blocks. Nodes that created blocks select random nodes in the block network and encrypt token information and own information with corresponding node's public key for transmission to the entire network. Nodes that have been deciphered with private key acquire the authority to create next block.

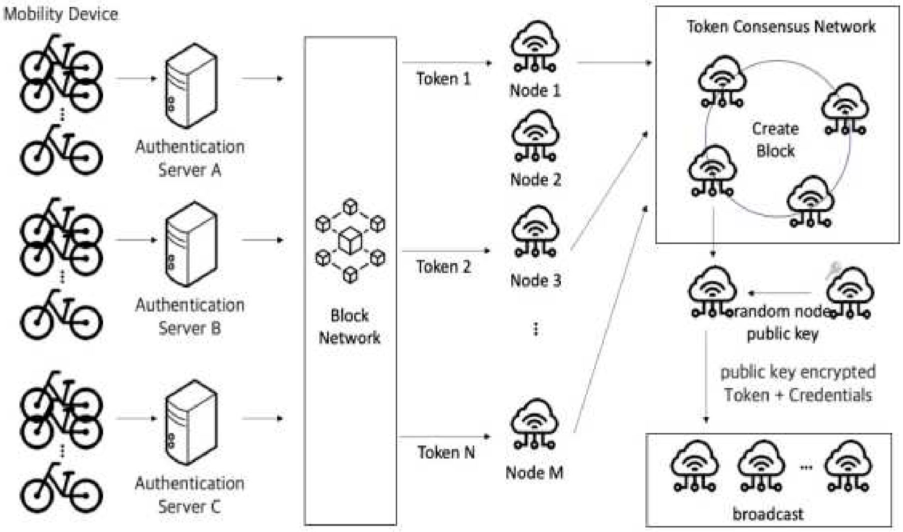

Figure 1. Conceptual drawing of block network random authentication node selection mechanism.

Block network random authentication node selection mechanism is divided into registration module that registered mobility devices on server, token issue module that creates token based on service provider's consensus and provides to block network, mobility transaction module that performs consensus process to process mobility transaction and token transmission module that transmits tokens to other nodes.

3.1. Mobility Device Registration Module

In mobility device registration module, mobility service provider performs device registration process via server, which handles provider's mobility device. In other words, block network's node is limited only to mobility devices authenticated by service provider. Information to recognize mobility device is required to register mobility device. [Table 2] is a list of data needed for the mobility device to be recognized in the proposed mechanism.

Table 2. Classification of mobility device data.

Data	Description
Identification no.(NID)	Device identification information created by service provider
MAC address(AM)	Physical address of mobility IoT network device
Service type(SC)	Type of mobility services provided
Service provider(SPV)	Service provider's identification information
Registration date(D)	Mobility information registration date

Authentication servers in block network identify devices by utilizing identification number created by service provider and MAD address of IoT communication device included in mobility device. Service provider creates device identification public information(DPI), which converted identification number and MAC address to hash value, and transmits public device consensus data(DPA) including service type, service provider and registration date to authentication server taking part in building mobility block network. [Formula 1] shows the process of creating hash value after performing hash computation of the two identification information to identify devices themselves. This is to keep too much information from being provided meaninglessly to other server that does not look for direct identification information. [Formula 2] means data structure to share information of mobility device to utilize nodes in other server.

$$D_{PI} = Hash(N_{ID} + A_M) \tag{1}$$

$$D_{PA} = str(D_{PI} + S_C + S_{PV} + D) \tag{2}$$

Authentication servers in block network that have public device consensus data check public device consensus data received and transmit number of mobility devices that have been requested to add in the block network, registration requested date and information of service provider to all authentication servers taking part in the block network. When identical information is shared across servers node creation authority whose number is the same as the aggregate of public device consensus data is added to the authentication server in the block network to which public device consensus data is transmitted. Authorized authentication server sets each mobility device as node and builds block network. [Figure 2] is the process of registering mobility devices into new nodes through consensus among authentication servers to have mobility devices participate in block network in the mobility device registration module.

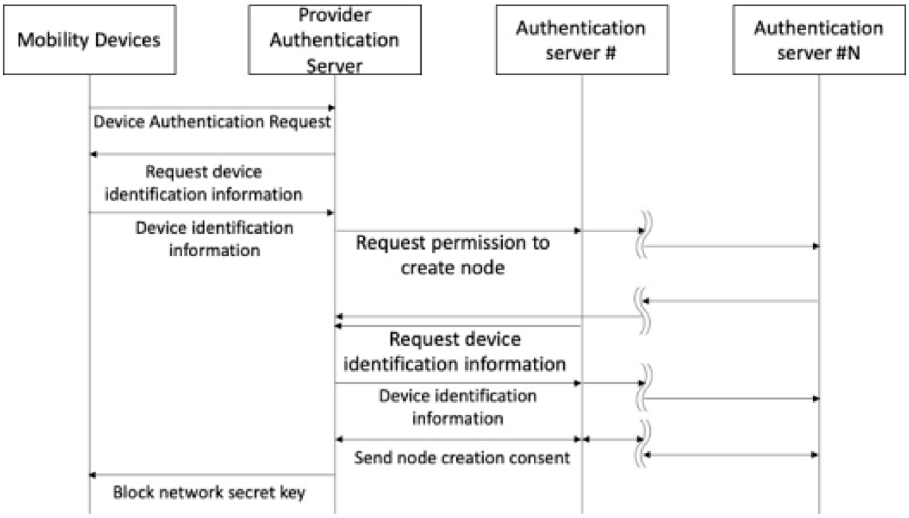
Mobility Devices
Provider Authentication Server
Authentication server #
Authentication server #N
Device Authentication Request
Request device identification information
Device identification information
Request permission to create node
Request device identification information
Device identification information
Send node creation consent
Block network secret key

Figure 2. Mobility device node registration process.

3.2. Token Issue Module

Token issue module is a module that issues token including authentication information and allows node that has a token on block network to create a block. Token initially issued is created based on consensus among authentication servers participating in block network. It includes matching key to grant authority on token issuance.

[Figure 3] is the configuration of token-created node transmitting token to random node belonging to block network. Token transmission requires two times of encryption. In the first one, the node, which created the token, encrypts token with its own secret key to inform that it has created a token. In the second one, encryption takes place with the target node's public key to make sure token information is not caught by node other than the target node that needs to have the token. As such, node without a token can confirm entire hash value, which is the correct answer to verify authority of the block, and question proving block creation authority, which is the question designed to verify authority. However, secret key on block creation is needed to solve question on proving block creation authority and hence verification cannot be completed. Node with a token can verify block with block creating secret key. [Table 3] shows information included in token.

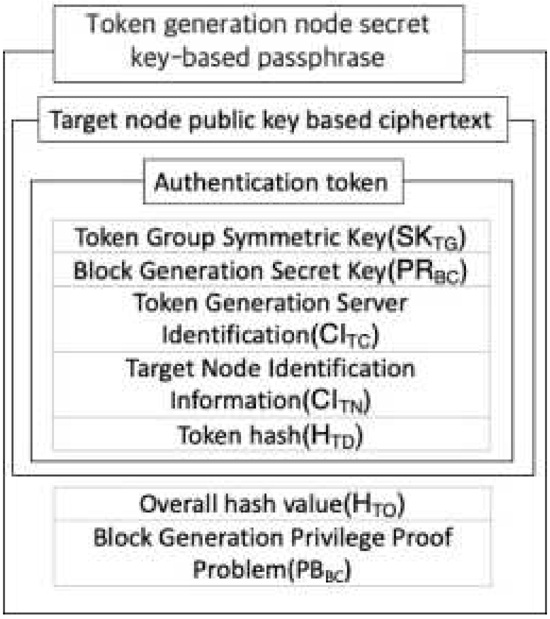
Token generation node secret key-based passphrase
Target node public key based ciphertext
Authentication token
Token Group Symmetric Key(SK_{TG})
Block Generation Secret Key(PR_{BC})
Token Generation Server Identification(CI_{TC})
Target Node Identification Information(CI_{TN})
Token hash(H_{TD})
Overall hash value(H_{TO})
Block Generation Privilege Proof Problem(PB_{BC})

Figure 3. Token transmission configuration.

Table 3. Token configuration.

Category	Description
Token group matching key(SK_{TG})	Code to build network between nodes with token
Block creation secret key(PR_{BC})	Answer to prove block creation authority
Token creation server identification information(CI_{TC})	Identification information of server that created a token
Target node identification information(CI_{TN})	Identification information of node to receive token
Token order information(R_{T})	Information marking token order
Token hash(H_{TD})	Hash computation value for all token characteristics

Token group matching key utilizes identification information of previous token group. When issuing token for the first time, information of token creation server that is in consensus with token group matching key is used.

[Figure 4] is the process of how a block creates tokens. Token that is initially issued includes matching key to build network of nodes that have tokens, secret key to be used as verification means to acquire authority to create next block, identification information of server that creates token, identification information of node that will receive token and lastly value that performed has computation for identification information of secret key and server, token order information and node identification. [Formula 4] shows the process of computing hash value for all token values.

$$SK_{TG} = Hash(\sum_{i=0}^{n} TokenID[i]) \tag{3}$$

$$H_{TD} = Hash(SK_{TG} + PB_{BC} + CI_{TC} + CI_{TN} + R_{T}) \tag{4}$$

When block creation secret key included in token is (d, N), block creation public key(PUBC) is (e, N) then token that has not been encrypted is configured as shown in [Formula 5].

$$Token\ cert = (Encryption\ Key_{TG}, Private\ Key(d, N), CI_{TC}, CI_{TN}, R_{T}, H_{TD}) \tag{5}$$

[Figure 5] shows the formula process to broadcast token encrypted by block creation node to the block network.

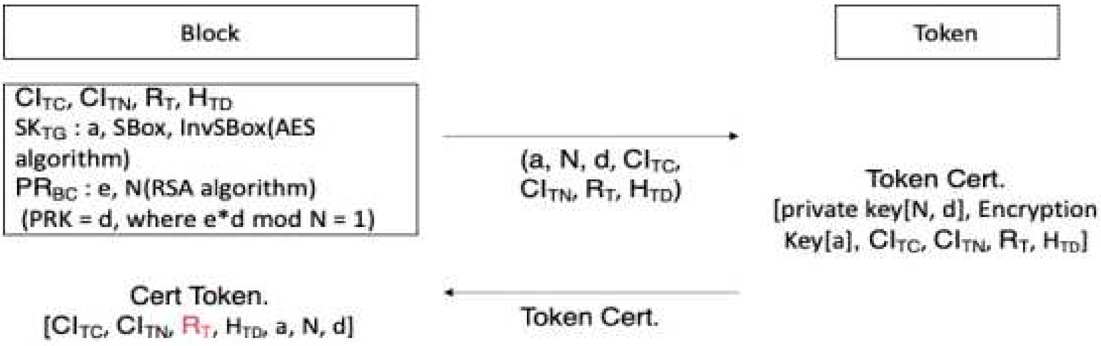
Block
Token
CI_TC, CI_TN, R_T, H_TD
SK_TG : a, SBox, InvSBox(AES algorithm)
PR_BC : e, N(RSA algorithm)
(PRK = d, where e*d mod N = 1)
(a, N, d, CI_TC, CI_TN, R_T, H_TD)
Token Cert.
[private key[N, d], Encryption Key[a], CI_TC, CI_TN, R_T, H_TD]
Cert Token.
[CI_TC, CI_TN, R_T, H_TD, a, N, d]
Token Cert.

Figure 4. Token creation process.

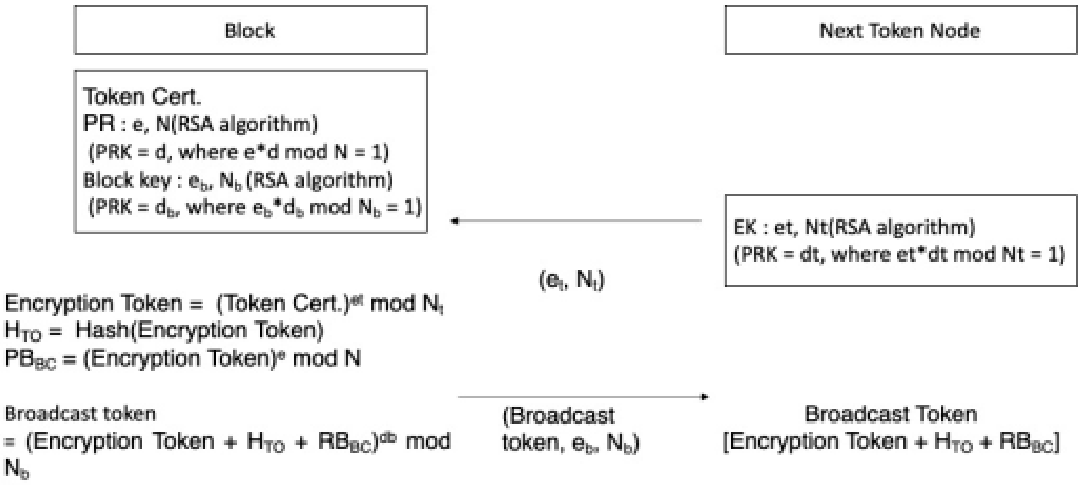
Block
Next Token Node
Token Cert.
PR : e, N(RSA algorithm)
(PRK = d, where e*d mod N = 1)
Block key : e_b, N_b (RSA algorithm)
(PRK = d_b, where e_b*d_b mod N_b = 1)
EK : et, Nt(RSA algorithm)
(PRK = dt, where et*dt mod Nt = 1)
(e_t, N_t)
Encryption Token = (Token Cert.)^et mod N_t
H_TO = Hash(Encryption Token)
PB_BC = (Encryption Token)^e mod N
Broadcast token = (Encryption Token + H_TO + RB_BC)^db mod N_b
(Broadcast token, e_b, N_b)
Broadcast Token
[Encryption Token + H_TO + RB_BC]

Figure 5. Process of creating question to prove block creation authority.

Token that has been created uses public key(et, Nt) of the node set to receive token to perform token encryption as shown in [Formula 6].

$$Encryption\ Token = (Token\ cert)^{et}\ mod\ N_t \tag{6}$$

Token is the created encrypted text. When it creates hash and block of encrypted token it encrypts with secret key of the node that creates token including block creation authority verification question(PBBC) to verify authority to create. [Formula 7] is the process to seek hash on the encrypted token and [Formula 8] is the process of how block creation authority is created by using the encrypted token.

$$T_{TO} = Hash(Encryption\ Token) \tag{7}$$

$$PB_{BC} = (H_{TO}\ mod\ N) \tag{8}$$

Token eventually broadcast to the block network is shown in [Formula 9].

$$Broadcast\ Token = (Encryption\ Token + H_{TO} + PB_{BC})^{db}\ mod\ N_b \tag{9}$$

[Figure 6] is the process of deciphering general nodes to which broadcast token has no authority over and tokens to node, which is the token's subject.

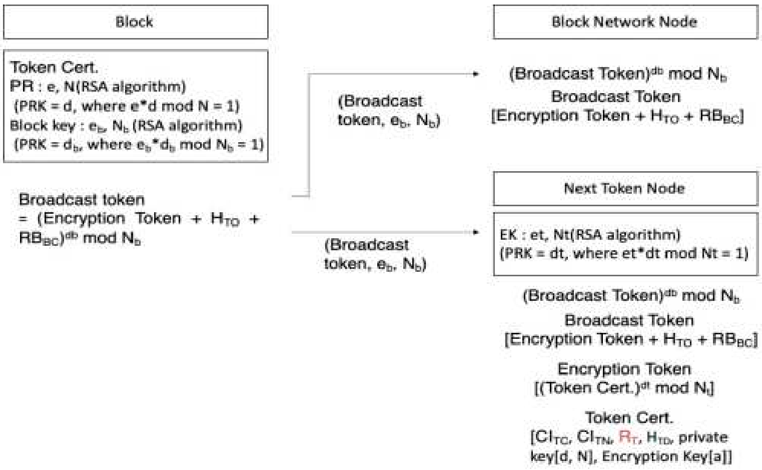
Block
Token Cert.
PR : e, N(RSA algorithm)
(PRK = d, where e*d mod N = 1)
Block key : e_b, N_b (RSA algorithm)
(PRK = d_b, where e_b*d_b mod N_b = 1)
Broadcast token
= (Encryption Token + H_{TO} + RB_{BC})^{db} mod N_b
(Broadcast token, e_b, N_b)
(Broadcast token, e_b, N_b)
Block Network Node
(Broadcast Token)^{db} mod N_b
Broadcast Token
[Encryption Token + H_{TO} + RB_{BC}]
Next Token Node
EK : et, Nt(RSA algorithm)
(PRK = dt, where et*dt mod Nt = 1)
(Broadcast Token)^{db} mod N_b
Broadcast Token
[Encryption Token + H_{TO} + RB_{BC}]
Encryption Token
[(Token Cert.)^{dt} mod N_t]
Token Cert.
[CI_{TC}, CI_{TN}, R_T, H_{TD}, private key[d, N], Encryption Key[a]]

Figure 6. Broadcast token deciphering process of general node and node subject to token.

Server that created token encrypts token by using public key of random nodes selected to transmit to random node in the block network and entire hash value and block creation authority verification question with its own secret key for broadcast to the block network. Node checks that it has been sent from the server by deciphering via server's public key and acquires encrypted token, entire hash value and block creation authority verification question. [Formula 10] shows the process of deciphering with public key sent from block to decipher broadcast token.

$$\begin{aligned} Plaintext\ Broadcast &= (Broadcast\ Token)^{eb} mod\ N_b \\ &= Encryption\ Token + H_{TO} + RB_{BC} \end{aligned} \tag{10}$$

Target node to which token should be sent uses its secret key to decipher token and acquire token group matching key in token and block creation secret key. [Formula 11] is the process of token target node deciphering encrypted token with its own secret key(dt, Nt) in order to decipher node.

$$\begin{gathered} Token = (Encryption\ Token)^{dt} mod\ N_t \\ = (Encryption\ key(SK_{TG}, Pravate\ Key(d, N), CI_{CT}, CI_{TN}, R_T, H_{TD}) \end{gathered} \tag{11}$$

Node creates token group by using matching key and collects transaction to create block. Nodes then create block and perform verification with its secret key. Nodes that failed to have token use secret key included in block verification to decipher block creation authority verification questions and check if the results are identical with the entire hash value to confirm verification. Nodes that have created block afterwards create new tokens by selecting secret key created by node, which has matching key and token that can be identically used among token groups, and random node to which token will be sent.

3.3. Mobility Transaction Module

Mobility transaction module is a series of process of collecting transactions and providing mobility service to users. It performs consensus process between transaction's data structure in the mobility node and transaction itself. Mobility transaction data is created by mobility device. It is executed when user information is forwarded to mobility device upon user request. Mobility device is composed of user information secured, additional time information, mobility device information and regional information, which serve as one single transaction. Mobility transaction data structure can be described in [Table 4].

Table 4. Mobility transaction data structure.

Data	Transaction	Description
User information	User identification information	Authentication information to identify hash-computed user
Time information	Time service starts	Time service starts as per request
Time information	Time service ends	Time service ends as per request
Regional information	Administrative district code	Administrative district identification code where mobility node is located
Regional information	GPS information	GPS information of mobility node
Mobility device information	Node identification information	Mobility device identification information
Mobility device information	Distance information	Specific user's distance information

Upon request for user by user mobility device fills out mobility transaction data including user information received upon request. The transaction is encrypted as mobility node's secret key and broadcast to the block network. The broadcast transaction then performs verification via token node.

[Figure 7] is the execution process to verify mobility transaction. All transactions are collected by token node and blocks are created based on consensus between token nodes.

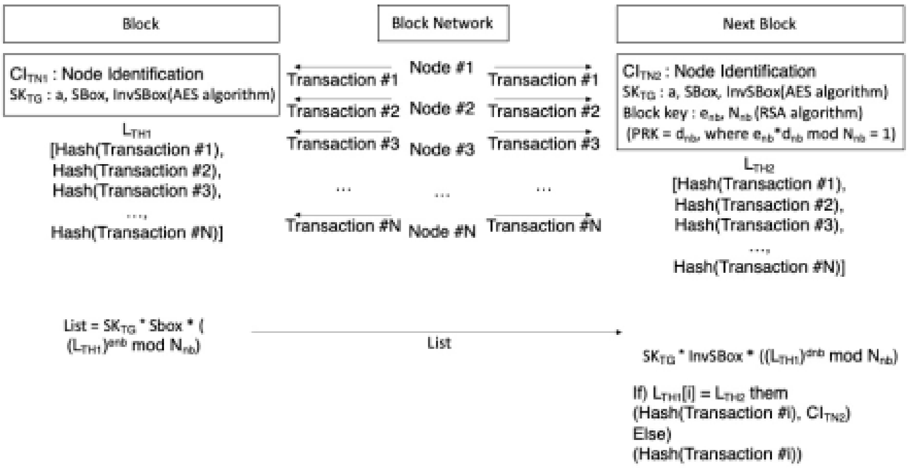
Block
Block Network
Next Block
CI_{TN1} : Node Identification
SK_{TG} : a, SBox, InvSBox(AES algorithm)
Transaction #1
Node #1
Transaction #1
Transaction #2
Node #2
Transaction #2
Transaction #3
Node #3
Transaction #3
...
Transaction #N
Node #N
Transaction #N
CI_{TN2} : Node Identification
SK_{TG} : a, SBox, InvSBox(AES algorithm)
Block key : e_{nb}, N_{nb} (RSA algorithm)
(PRK = d_{nb}, where e_{nb}*d_{nb} mod N_{nb} = 1)
L_{TH1}
[Hash(Transaction #1),
Hash(Transaction #2),
Hash(Transaction #3),
...,
Hash(Transaction #N)]
L_{TH2}
[Hash(Transaction #1),
Hash(Transaction #2),
Hash(Transaction #3),
...,
Hash(Transaction #N)]
List = SK_{TG} * Sbox * ($(L_{TH1})^{enb}$ mod N_{nb})
List
SK_{TG} * InvSBox * ($(L_{TH1})^{dnb}$ mod N_{nb})
If) $L_{TH1}[i] = L_{TH2}$ them
(Hash(Transaction #i), CI_{TN2})
Else)
(Hash(Transaction #i))

Figure 7. Mobility transaction verification data broadcast process.

The collected transaction is collected by token node and hash computation is performed for each transaction to create hast list(LTH). [Formula 12] is the process whereby hash list is created.

$$L_{TH} = \sum_{i=0}^{n} Hash(Transaction[i]) \tag{12}$$

The first token node creates transaction hash list(LTH), which is the result of executing hash computation for the transaction collected. Transaction hash list that has been created is encrypted as the public key(PUTN) of token node following in the next order and is encrypted as token group matching key for broadcast to block network. When public key is (enb, Nnb) and secret key is (dnb, Nnb) for node that will receive transaction hast list [Formula 13] broadcasts transaction hash list to the block network to show the process of deciphering token group to a certain node. [Formula 14] shows token group encryption when secret key of token group is a and SBox for encryption and matching table for deciphering is InvSBox.

$$Next\ Token\ Node\ Encryption = ((L_{TH})^{enb}\ mod\ N_{nb}) \tag{13}$$

$$Token\ Group\ Encryption = a * SBox * ((L_{TH})^{enb}\ mod\ N_{nb}) \tag{14}$$

The broadcast transaction list is deciphered first by token group's matching key and thus is shared only within token group. Even within the group the transaction hash list as it is can be obtained only with a certain node's secret key. [Formula 15] is the process of a certain selected node executing deciphering via token group matching key. [Formula 16] is the process of acquiring transaction hash list by using own secret key.

$$\begin{aligned} 1st\ Decryption &= a * InvSBox * ((L_{TH})^{enb}\ mod\ N_{nb}) \\ &= (L_{TH})^{enb}\ mod\ N_{nb} \end{aligned} \tag{15}$$

$$\begin{aligned} 2st\ Decryption &= (L_{TH})^{dnb}\ mod\ N_{nb} \\ &= L_{TH} \end{aligned} \tag{16}$$

Token node compares broadcast transaction hash list with its own transaction hash list to perform verification with its own identification information for the same items. It then broadcasts transaction hash list verified to the next token node including verification result of transaction hash list. [Formula 17] is the process of executing verification by comparing transaction hash list with own transaction has list and performing renewal with verification result including its own identification information.

$$Transaction\ List = \begin{cases} Hash(Transaction[i]), CI_{TNx}, if) Hash(Transaction[i]) \in L_{THx+1} \\ \quad Hash(Transaction[i]), else) Hash(Transaction[i]) \notin L_{THx+1} \end{cases} \tag{17}$$

The renewed transaction list repeats the process from [Formula 12] to [Formula 14] before sending to the next block.

[Figure 8] is the general consensus process of token nodes to perform transaction verification. Token node that has received verified transaction hash list of all token nodes creates transaction verification list(LTV) including transaction list verified by all token nodes and transaction's hash verification list verified by more than half of nodes. It is then encrypted with token group matching key and broadcast to all networks. This is followed by request for verification and its propagation to the incoming node. Token node that has been requested compares it with its own transaction list and sends it to all token groups when there exists transactions with multiple verifications(TDV). Transactions not in the transaction list remove themselves in their own list and renew order as per the list before propagating request to perform verification to the node next in order. The first node requested for transaction verification by Nth token node verifies its own block with the secret key in the authentication token and broadcasts to the node next in order. Token nodes that received block in the consecutive order then verify with its own secret key. Nth token node broadcasts to the block network to let block verification by all nodes when number of verifies is more than half of the entire token group.

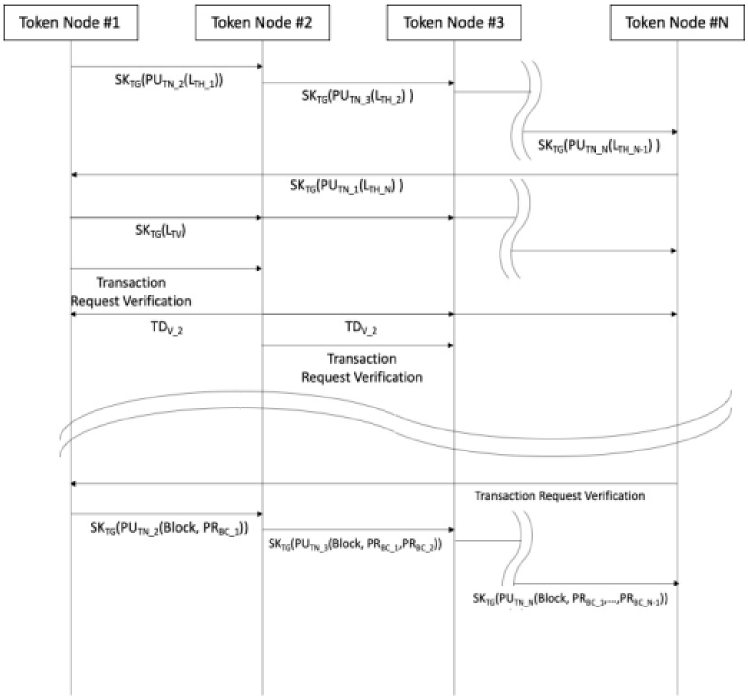
Token Node #1
Token Node #2
Token Node #3
Token Node #N
SK_TG(PU_TN_2(L_TH_1))
SK_TG(PU_TN_3(L_TH_2))
SK_TG(PU_TN_N(L_TH_N-1))
SK_TG(PU_TN_1(L_TH_N))
SK_TG(L_TV)
Transaction Request Verification
TD_V_2
TD_V_2
Transaction Request Verification
Transaction Request Verification
SK_TG(PU_TN_2(Block, PR_BC_1))
SK_TG(PU_TN_3(Block, PR_BC_1,PR_BC_2))
SK_TG(PU_TN_N(Block, PR_BC_1,...,PR_BC_N-1))

Figure 8. Transaction verification process by token node.

3.4. Token Transmission Module

Token transmission module's role is to create block, issue new token and transmit them to the next block. Token group node uses token's block creation secret key to verify block verification in order to create block. It then issues and attaches token from among the next node that has been randomly selected. The newly-issued token creates token group matching key, block creation secret key, token creation server identification information, target node identification information and token hash identical with the authentication token created by server.

Token group matching key creates one matching key via token node group and each token node uses matching key created to its own token group matching key, which is then used as the matching key for the next node to form token group. Since block creation public key pair is the solution to prove itself to create block each token node creates independent public key pair with different contents. Token creation server identification information enters identification information of node renewing token and used to prove node that has been created by random node receiving token. Target node identification information refers to identification information of random node and token hash adds hash value that performed computation towards token just like the server. In short, token node that created a block itself works as a single CA(Certificate Authority) and plays the role of a one-off certifying agency for token.

[Figure 9] is token verification process. Random node that acquired token information confirms node that created token via token creation server identification information and encrypts its own identification information authentication token's token hash with its public key for propagation(broadcasting). By using its secret key, verification node identifies authentication request by the encrypted random node and encrypts identification information of verification node and verification findings with random node public key for broadcast to the block network. Random node uses token as per verification results and creates token node group.

Token that has been created perform encoding with the public key of randomly selected node to prevent token details being restored by a node other than the target node. It then performs encoding with secret key of node that creates token to have token issuing entity be verified on the block network. The encrypted token is then broadcast to the entire block network. Block network nodes that receive encrypted tokens utilizer public key to get encrypted token information, all hash values and block creation authority verification questions, and uses its own secret key to acquire token information. When token information is restored by its own secret key it uses token group matching key to create token node group and token node creates block by verifying transactions.

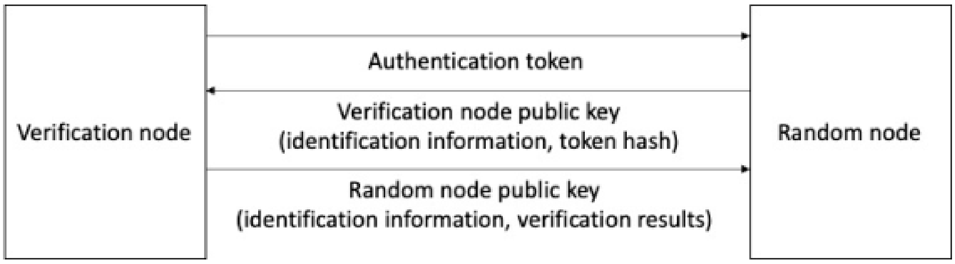
Verification node
Authentication token
Verification node public key
(identification information, token hash)
Random node public key
(identification information, verification results)
Random node

Figure 9. Token authentication process.

4. Comparative Analysis with Existing Methodology

This thesis is proposing to select node with block creation authority on the block network based on random token and mechanism for processing mobility transactions. The process of consensus on block creation by using block creation authority requires comparative analysis with the consensus mechanism. The most common method for consensus mechanism is PoW and PoS while DPoS and PoA are also available.

In PoW consensus mechanism, which is the most known method, node computes hash value by substituting nonce value, which grows by one, to solve the question once node that created block broadcasts questions to all nodes. Result of the computed hash has to look for a hash that is smaller than the question and node that found the most number of values acquire the authority to create block and compensation by proving that it provided a lot of resources to the block network. Since block creation authority is gained by solving many questions it is suitable for decentralization since all nodes have the authority to create blocks but more competition associated with problem-solving wastes too much energy. As such, it is not an ideal choice for network based on low-performance IoT environment.

PoS consensus mechanism prevent inefficient energy consumption arising from intensifying competition by PoW consensus mechanism. It is also used to prevent monopoly by certain nodes that provide most amount of resources. What is used to prove block in PoS method is shares held by nodes not computing power by high performance devices to solve problems. Random verifier is selected to create block in PoS and the chances of being selected go up depending on how many tokens can be put up for stakes for block creation. Although it is a random selection it indicates that higher token stakes raises the chance for block creation. In short, the more the stakes by a node, the higher the chances of winning block creation authority. New nodes face higher entry barrier since they have less stakes than the existing ones, which could cause issues around fixation of block creation nodes.

DPoS consensus mechanism is one proposed to address the limitations of certain nodes monopolizing block creation authority by the stakes in PoS method. In DPoS method, which selects proxy to create block, each block selects proxy as per its own stakes. Its processing speed is fast since only a few selected proxy nodes verify transaction records. This isn't without downsides. Two nodes with many stakes vote against each other and take a fixated form in order to stay qualified as block creation proxy.

Random token-based selection method proposed in this thesis allows only nodes that have token to take part in the group for block creation and creates consensus blocks in the token group network. As for nodes authorized to create block random nodes on the block network are selected and therefore all nodes have a chance to take part in block creation. Also, it does not require high performance since there does not need a computation process to acquire block creation authority. However, nodes authorized to create blocks maintain a consistent number and there is a risk of block forgery/counterfeit if more than half of selected nodes have malign intentions. Also, an issue of redundancy where more than two token nodes can set one single node into the next token node exists.

[Table 5] shows the difference in the consensus between the existing consensus mechanism and proposed mechanism. PoW needs to perform the biggest workload for block creation authority. In

PoS node with biggest stakes has higher chance of acquiring block creation authority. In DPoS, stakes may be the key role in participating in block creation but it is different in that stakes are used to select proxy. The proposed mechanism utilizes token broadcast to the randomly-selected node in selecting proxy. High-performance mining is required only in PoW and not in PoS, DPoS and the proposed mechanism. Node itself is the verifier in PoW and PoS in that key node is selected to create block while key node in DPoS and the proposed mechanism is determined by vote based on each stake and random selection. PoW proof can be executed anonymously but node information is required for PoS and DPoS when selecting stakes or proxy. In the proposed mechanism token group matching key is used to let node with token only take part in the network. As such, node without token cannot gain token group information.

Table 5. Comparative analysis of existing consensus mechanism and proposed mechanism.

	PoW	PoS	DPoS	Proposed mechanism
Block creation authority	Work	Stake	Stake	Token
Mining	O	X	X	X
Required computation power	High	Low	Low	Low
Proxy node selection	X	X	O	O
Anonymity	O	X	X	O

5. Conclusions

Advances in small IoT devices and growing demand for electric power devices are bringing major changes in the size of mobility services. In vehicles, physical power devices for control as an independent entity is giving way to sensor-controlled vehicle function applied with intelligent vehicle technologies to provide user convenience. Even small mobility devices like kick-boards and bicycles are seeing a shift from human-driven power to power device based on electric batteries for mobility service. This suggests that mobility service is relying more on intelligent mobility devices and control increasingly based on sensor.

Intelligent control based on sensor can provide convenience to mobility service user but it runs the risk of ill intentions by forging data or maliciously counterfeiting mobility service data to harm the mobility big data environment. Attacks when controlling intelligent vehicle via sensor communication or communicating with shared mobility services need to be prevented. At the same time a verification means to build a reliable mutual communication is needed and this is where block chain is coming into play as a solution.

This thesis proposed a mechanism where randomly selects token nodes by using a token, which creates blocks in the block network that has been built by a device authenticated by restricted authentication server in order to apply block chain in the mobility environment. By selecting random node as node to create block, block creation fixated by a certain node can be prevented. Also, token node information is not provided to another node since node that has been transmitted with token can decipher token information only with its secret key. Future studies need to focus on addressing redundant selection arising from selecting node and solutions to mitigate required performance by lighter encryption.

Author Contributions: J.K. contributed significantly to the conceptualization of the mechanism and the writing of the manuscript. J.K is currently working as a researcher at the Cybersecurity Human Resource Institute, Jeju National University. His current research interest includes Cloud System, Intelligent Video Surveillance System and Internet of Things.; E.C. contributed significantly to the revision of the manuscript. E.C is currently working as a researcher Creative Education Base Center, Jeju National University. His current research interest includes Computer Education, STEAM, Digital Humanities.; B.K. contributed significantly to the revision of the manuscript. B.K is currently working as a Professor in the Dept. of Artificial Intelligence Engineering, Sookmyung Women's University. His current research interest includes Image and Video Signal Processing Computer Vision Deep Learning Pattern Recognition Video Coding Standards.; N.P. contributed greatly to the conceptualization of methodology and overall project management and preparation of manuscripts. All authors have read and agreed to the published version of the manuscript. N.P is currently working as a Professor in the

Department of Computer Education, Teachers College, Jeju National University. His current research interest includes Convergence Technology Security, Computer Education, Smart Grid, IoT, Sea Cloud.

Funding: This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2022S1A5C2A04092269). And, this work was supported by Electronics and Telecommunications Research Institute(ETRI) grant funded by the Korean government. [23ZD1160, Regional Industry ICT Convergence Technology Advancement and Support Project in Daegu-GyeongBuk(Mobility)]

Acknowledgments: Thanks to the editors and reviewers.

Conflicts of Interest: The authors declare that they have no competing interests.

References

1. Hao Z.; Ka-Veng Y.; Lyudmila M.; Henry L. Overview of Environment Perception for Intelligent Vehicles. IEEE Transactions on Intelligent Transportation Systems 2017, 18(10), 2584-2601.
2. Alberto B.; Alex Z.; Ümit Ö.; Christian L. Intelligent Vehicles. Springer Handbook of Robotics 2016, 1627-1656.
3. Rob J.F.; Jana S.; Jörg W.; Andreas H.; Florian U.; Markus B.; Helmut K. Introducing platform ecosystem resilience: leveraging mobility platforms and their ecosystems for the new normal during COVID-19. European Journal of Information Systems 2021, 30(3), 304-321.
4. Francesco F.; Guido P.; Mariangela R.; Andrea V. Car-sharing services: An annotated review. Sustainable Cities and Society 2018. 37, 501-518.
5. Chinh Q.H.; Corinne M.; David A.H. Public preferences for mobility as a service: Insights from stated preference surveys. Transportation Research Part A: Policy and Practice 2020, 131, 70-90.
6. Pierre K.; Tomas O.; Erland J. Security aspects of the in-vehicle network in the connected car. 2011 IEEE Intelligent Vehicles Symposium (IV) 2011.
7. Namje P.; Namhi K. Mutual Authentication Scheme in Secure Internet of Things Technology for Comfortable Lifestyle. Journal of Sensors (Basel) 2015, 16(1), 1-16.
8. Joseph A.; Mohammad E.K.; Bassam M.; Ribal A.; Chadi A. A Detailed Security Assessment of the EV Charging Ecosystem. IEEE Network 2020, 34(3), 200-207.
9. Zhang .; Luo L.; Li R.; Yi J.; Li Y.; Chen L. Research and Application of Intelligent Vehicle Cybersecurity Threat Model. 2022 7th IEEE International Conference on Data Science in Cyberspace (DSC) 2022.
10. Erik K.; Marvin J.; Frederik A. Blockchain for Smart Mobility—Literature Review and Future Research Agenda. Sustainability 2021, 13(23).
11. Bulat N.; Muhammad M.; Qiang Q. ChainMOB: Mobility Analytics on Blockchain. 2018 19th IEEE International Conference on Mobile Data Management (MDM) 2018.
12. Jinsu K.; Namje P. Lightweight knowledge-based authentication model for intelligent closed circuit television in mobile personal computing. Personal and Ubiquitous Computing 2019, 1-9.
13. Sara P.; Mohd A.A.; Sherin Z.; Gautami T.; Aqeel K.; Imran H. Privacy and security challenges in smart and sustainable mobility. SN Applied Sciences 2020, 2(1175).
14. Radosław W. Analysis of the Bicycle Roads System as an Element of a Smart Mobility on the Example of Poland Provinces. Smart Cities 2023, 6(1), 2023.
15. Sanja Š.; Tullio G.; Aleksandra D.T. Smart mobility solutions – necessary precondition for a well-functioning smart city. Transportation Research Procedia 2020, 45, 604-611.
16. Namje P.; Byung-Gyu K.; Jinsu K. A Mechanism of Masking Identification Information regarding Moving Objects Recorded on Visual Surveillance Systems by Differentially Implementing Access Permission. ELECTRONICS 2019, 8(7), 735.
17. Fahad S. Blockchain without Waste: Proof-of-Stake. The Review of Financial Studies 2021, 34(3), 1156–1190.
18. Cristian L.; Michela C.; Andrea V.; Udai P.R.; Kaushal A.S.; Luca Z. A Survey on Blockchain Consensus with a Performance Comparison of PoW, PoS and Pure PoS. Mathematics 2020, 8(10), 1782.
19. Benny V.; Mads L.; Huan N.; Istvan Z.K.; Preben M.; Mads S. Coverage and Capacity Analysis of Sigfox, LoRa, GPRS, and NB-IoT. 2017 IEEE 85th Vehicular Technology Conference (VTC Spring) 2017.
20. Mohieddine E.S.; Pouria Z.; Frank P.; Guido D. Evaluating the Performance of eMTC and NB-IoT for Smart City Applications. 2018 IEEE International Conference on Communications (ICC) 2018.
21. Aloÿs A.; Jiazi Y.; Thomas C.; William M.T. A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. Sensors 2016, 16(9), 1466.
22. Rapeepat R.; Benny V.; Nitin M.; Amitava G. NB-IoT system for M2M communication. IEEE Wireless Communications and Networking Conference 2016.
23. Zhirong Z.; Xuetian Z.; Zhijun L.; Yong Z. Analysis of the Impact of eMTC on Legacy LTE. 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC) 2019.

24. P. Rajitha N.; D. Ramya D. Evaluation of Performance and Security of Proof of Work and Proof of Stake using Blockchain. 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV) 2021.
25. Namje P.; Jungsoo P.; Hyoungjun K. Inter-Authentication and Session Key Sharing Procedure for Secure M2M/IoT Environment. International Information Institute(Tokyo) Information 2015, 18(1), 261-266.
26. Shihab S.H.; Qusay H.M. Improving Transaction Speed and Scalability of Blockchain Systems via Parallel Proof of Work. Future internet 2020, 12(8), 125.
27. Congcong Y.; Guoqiang L.; Hongming C.; Yonggen G.; Akira F. Analysis of Security in Blockchain: Case Study in 51%-Attack Detecting. 2018 5th International Conference on Dependable Systems and Their Applications (DSA) 2018.
28. Namje P.; Hyochan B. Mobile middleware platform for secure vessel traffic system in IoT service environment. Journal of Security and Communication Networks 2014, 500-512.
29. Christophe S. Proof-of-work based blockchain technology and Anthropocene: An undermined situation?. Renewable and Sustainable Energy Reviews 2021, 152.
30. Donghyeok L.; Namje P. Geocasting-based synchronization of Almanac on the maritime cloud for distributed smart surveillance. Supercomputing 2017, 73(3), 1103-1118.
31. Sriman B.; Ganesh K.S.; Shamili P. Blockchain Technology: Consensus Protocol Proof of Work and Proof of Stake. Intelligent Computing and Applications 2020, 1172, 395-406.
32. Alex D.V. Cryptocurrencies on the roadto sustainability: Ethereum pavingthe way for Bitcoin. Patterns 2023, 4(1).
33. Elie K.; Bruce M. An Event Study of the Ethereum Transition to Proof-of-Stake. Commodities 2021, 2(2), 96-110.
34. Qian H.; Biwei Y.; Yubing H.; Jiguo Y. An Improved Delegated Proof of Stake Consensus Algorithm. Procedia Computer Science 2021, 187, 341-346.
35. Jinsu K.; Namje P. Blockchain-Based Data-Preserving AI Learning Environment Model for AI Cybersecurity Systems in IoT Service Environments. Applied Sciences 2020.
36. Sophia A.; Sophia N.; Somnath M.; Raghava R.M. Towards blockchain-IoT based shared mobility: Car-sharing and leasing as a case study. Journal of Network and Computer Applications 2022, 200.
37. Madhusudan S.; Shiho K. Blockchain Based Intelligent Vehicle Data sharing Framework. Cryptography and Security 2017.
38. Madhusudan S.; Shiho K. Branch based blockchain technology in intelligent vehicle. Computer Networks 2018, 145, 219-231.
39. Hanyue G.; Jiting Z.; Jiaqi W.; Xiaodong W. A bike sharing system based on Blockchain platform. 2018 2nd International Conference on Electronic Information Technology and Computer Engineering (EITCE 2018) 2018.
40. Daozhi Z.; Di W.; Baosen W. Research on a Shared Bicycle Deposit Management System Based on Blockchain Technology. Journal of Advanced Transportation 2020, 2020, 1-14.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.