Network Slicing Handover Authentication: A Hyperledger Fabric-Driven Privacy and Security Solution

Igugu Tshisekedi Etienne[1], Cho Nwe Zin Latt[1], Muhammad Firdaus[2], and Kyung-Hyune Rhee[3]

[1] Department of Information Security, Pukyong National University, Busan, South Korea
{iguguetienne, chocho1612}@pukyong.ac.kr

[2] Department of Artificial Intelligence Convergence, Pukyong National University, Busan, South Korea
mfirdaus@pukyong.ac.kr

[3] Division of Computer and Artificial Intelligence Engineering, Pukyong National University, Busan, South Korea
khrhee@pknu.ac.kr

Abstract

The proposed research aims to leverage blockchain technology to enhance the security of HA processes within network slicing systems. It is achieved by establishing a decentralized and secure registry for recording transfer events. In this sense, the simplification of device identification verification leads to an improvement in Handover authentication (HA) efficiency. Furthermore, our work introduces a three-component model: network slicing, user environments, and a Hyperledger Fabric (HLF) blockchain dedicated to authentication and authorization. We use HLF thanks to its advanced privacy and security features, which promise to enhance user experience during handover by minimizing delays, ensuring data privacy, and providing flexibility and scalability. Additionally, we conducted several experiments to facilitate the assessment of system performance results. The HLF network performance analysis reveals that while the average transaction time and request response time show a slight increase with the growing number of users, there is a significant improvement in the average network throughput. It indicates the network demonstrates satisfactory scalability, maintaining acceptable performance under increasing load. Therefore, the results represent that the underlying system, based on HLF, is suitable for effectively managing network scenarios while maintaining service quality and network security. In essence, this work enhances network security and heightens service quality, particularly within network slicing, HA, and adopting a privacy and security solution based on HLF.

Keywords: 5G, network slicing, handover authentication, hyperledger fabric, identity mixer

1 Introduction

In a network slicing system, Handover Authentication (HA) involves verifying a device's identity during its transition from one network slice to another. It is crucial to prioritize network slices based on their varying security requirements and access policies. HA is critical in implementing network slicing in 5G networks. Through network slicing, multiple virtual networks can be established on a single physical infrastructure, each tailored to specific use cases with unique demands for security, performance, and quality of service. Ensuring secure handovers among these virtual networks addresses potential security threats and prevents service disruptions. For instance, when a device seeks access to a new slice, it presents a blockchain-based token containing information about its identity and the slice which is permitted to access. The network

The 7th International Conference on Mobile Internet Security (MobiSec'23), Dec 19-21, 2023, Okinawa, Japan, Article No.54

slice can verify the token's authenticity through the blockchain and then approve or reject access accordingly. Adopting a blockchain for HA has several benefits. Being decentralized and secure, it can offer an unalterable record of device identities and access rights, ensuring that only authorized devices can access the network slices to which they are entitled. Furthermore, using a blockchain for HA can simplify the verification of device identities, eliminating the requirement for a centralized authority to oversee the authentication process.

The regulation and standardization of blockchain for HA in network slicing continue to evolve, promising to maintain the maintenance of handover and network management to improve performance and authentication within the network. This integration takes place within the existing network infrastructure, ensuring a seamless and efficient transition to meet the growing demand for network slicing and handover. However, such adoption necessitates a widespread and meaningful understanding of blockchain technology and a clear comprehension of the investment's benefits. The research aims to leverage blockchain technology for storing and verifying device identities during their movement within and across network slices, establishing a decentralized and tamper-resistant record of events. Hence, it enhances HA within network slicing. On the other hand, involving blockchain in cellular networks builds trust and security during user connection transitions, offering undeniable identity verification and maintaining data integrity. Thus, the secure storage and sharing of mobile user identity, network settings, and handover information through blockchain strengthen overall connectivity efficiency and speed. In summary, the contributions of this paper are:

- Our proposed system verifies UE identities throughout the handover process during network slicing by leveraging the advantages of the HLF platform.

- The system prioritizes security and privacy protection by ensuring transparency and limiting access only to authorized users.

- Through the integration of IM, the system provides anonymity and facilitates flexible data management for channel stakeholders.

- We conducted simulations to analyze system performance based on several metrics, i.e., transactions per second (TPS), latency, and throughput.

Our remaining work is organized as follows: Section 2 and Section 3 offer an overview of the existing system incorporating blockchain technology as an authentication solution in a layered network handover process. Section 4 introduces our proposed system, emphasizing its capacity to adapt and regulate the blockchain authentication solution to address the growing demand for network slicing and handover. This section also highlights the performance analysis of our implementation using HLF during the handover in network slicing. Finally, in Section 5, we present our conclusion.

2 Literature Review of Network Slicing and Handover Process

This chapter explores the relationship between network slicing and the handover process, highlighting how they collaborate to provide access to a network slice's services. It offers an overview of the current state of network slicing, handover processes, and authentication.

2.1 Network Slicing Management

In 5G and beyond networks, the terms of network slicing customize virtual networks for specific applications like IoT, autonomous vehicles, or critical services. This technique divides a physical network into distinct virtual networks, each with unique capabilities. Measures include latency, throughput, error rate, reliability, availability, and functional aspects like mobility, security, and control. These slices are created on demand, operated independently, and managed separately. Proper isolation is crucial to preventing interference. On the other hand, the design management of communication networks integrates various technologies, such as network slicing, software-defined networks (SDN), and network function virtualization (NFV). Network slicing deploys tailored virtual networks for specific applications, optimizing resource management and enabling flexible, automated orchestration [1], [2]. The architecture involves interconnected layers: infrastructure for physical components, virtualization for isolated networks, element managers for performance and data collection, and virtualized network functions (VNFs) for traffic management [3].

2.1.1 Slices

In wireless networks like 5G and beyond, ”slices” refer to specialized segments known as network or service slices. These are dedicated portions of the network designed to meet specific application requirements. Network slicing partitions the same physical infrastructure into multiple virtual networks, each optimized for a particular use case, supporting various applications with diverse demands. For example, slices can be tailored for enhanced mobile broadband (eMBB), massive machine-type communication (mMTC), and ultra-reliable low-latency communication (URLLC) [4]. The network slicing layer (NSL) interacts with the SDN Controller Layer to allocate necessary resources for creating a slice [5]. On the other hand, Single Network Slice Selection Assistance Information (S-NSSAI) shows that each UE can be associated with up to eight slices simultaneously. It is an essential part of the non-access stratum 5G mobility management (NAS 5GMM) framework, which is made up of the slice type (SST) to describe the attributes and functions of a slice and the slice differentiator (SD) to tell slices in the same SST category apart. While SST is mandatory, SD supports differentiating slices with identical SST [6].

2.1.2 Initial Registration Request

Reliable identification of User Equipment (UE) in network slicing is crucial for secure resource allocation and maintaining service quality. Each UE is associated with a specific network slice based on identifiers like the third-party slice identifier and the UE identifier [7]. This information ensures precise matching between network slices and terminals, directing them to the appropriate slice with the desired service type [8]. During initial registration in a 5G network, a UE establishes its identity through a Registration Request, transmitting vital information like IMSI (international mobile subscriber identity) or alternative identifiers (e.g., TMSI or 5G-GUTI). This forms a secure and unique connection between the UE and the network. The Registration Request, part of NAS-5GMM, carries details about the UE’s capabilities and supported network features, including the Requested Network Slice Selection Assistance Information (NSSAI) parameter, allowing the UE to express its slice preferences to the network. These preferences can dynamically adapt based on device configuration or network policies from the PCF (policy control function), optimizing resource usage and enhancing the user experience.

In Figure 1, UE transmits its identifier and NSSAI during registration. NSSAI specifies

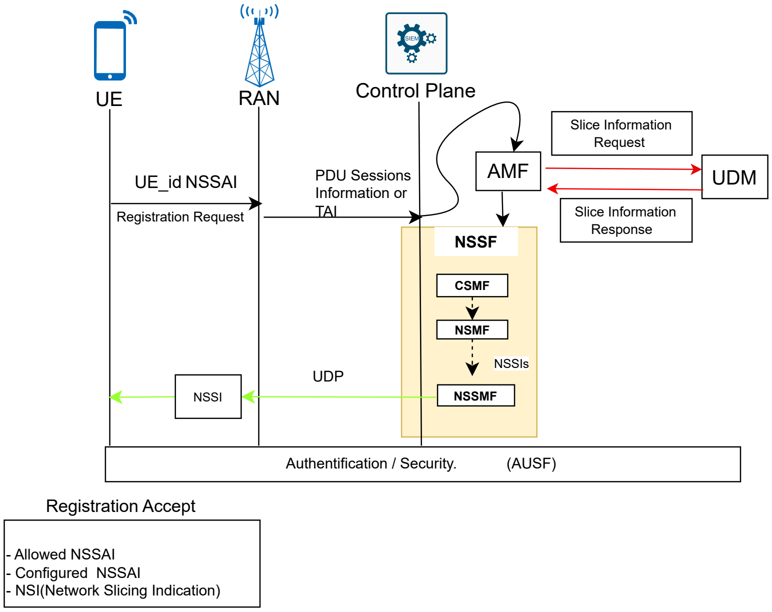

UE
RAN
Control Plane
UE_id NSSAI
Registration Request
PDU Sessions Information or TAI
AMF
Slice Information Request
UDM
Slice Information Response
NSSF
CSMF
NSMF
NSSIs
NSSMF
UDP
NSSI
Authentification / Security. (AUSF)
Registration Accept
- Allowed NSSAI
- Configured NSSAI
- NSI(Network Slicing Indication)

Figure 1: Registration process

desired slice types. This data is sent to UDM (unified data management) for validation. If authorized, the user datagram protocol (UDP) accepts and notifies UE, providing additional information like configured NSSI and inclusion mode [9]. If rejected, the request is denied. Post-acceptance, the core network undergoes authentication and verification steps by the authentication server function (AUSF), ensuring communication confidentiality and security. This process validates UE legitimacy and authorization for slice access, enhancing data protection and services in 5G.

2.2 Handover Process in a Sliced Network

The handover process in 5G services has different requirements, including seamless session continuity, minimal latency, support for high user mobility, and maintaining a constant data rate while minimizing re-transmissions [10]. The HA process implemented within network slicing aims to guarantee the security and integrity of the handover process, prohibit any unauthorized network access, and safeguard the confidentiality of user data.[11]

Handover Decision (HO Decision) : The handover (mobility) of UE within network slices is primarily triggered by events occurring within a network slice. These events can include factors such as those mentioned above, leading to a degraded or non-optimized connection for a UE. When such an event occurs, based on the UE's requirements, the network through the base station can determine if a handover is necessary and to which network slice the UE should be transferred [12].

HO Decision Algorithms : It is a method to determine the best course of action to allow UE to move between different networks for resource optimization. Several approaches are related to handover decision, such as the multi-criteria vertical handoff decision algorithm for overlaid heterogeneous mobile networks proposed by Farouk M. et al. in [13], which uses multiple criteria, such as received signal strength (RSS), traffic class, and speed for decision making. Another approach proposed in [14] is to use machine learning techniques, such as Q-learning or fuzzy logic, to predict if a handover is necessary and make decisions.

In the process of UE registration, several functional steps are involved. A configured NSSAI is initially provided to the UE based on the network slice selection policy (NSSP). The UE then sends a Registration Request, including the Requested NSSAI, to the radio access network (RAN). The RAN selects the access and mobility management function (AMF) based on the Requested NSSAI and other criteria, performing necessary security procedures. In handover cases, the allowed NSSAI includes mapping each selected NSSAI to the values the UE subscribes to in the home public land mobile network (HPLMN). The AMF fetches subscription data from the UDM, which returns both Subscription data and Subscribed NSSAI. Following this, the AMF interrogates the NSSF for slice selection, receiving a set of network slice instance IDs and allowed S-NSSAI. Finally, the AMF sends a registration accept/complete message, including allowed S-NSSAIs, 5G-GUTI (globally unique temporary identifier), Registration area, mobility restrictions, and more, to both the RAN and UE through the NAS. Subsequently, the UE uses the updated NSSAI, providing a list of allowed S-NSSAIs for further communication (handover).

3 Fundamentals of Blockchain and Hyperledger Fabric

Blockchain, a decentralized ledger, ensures high transparency and security in recording transactions and tracking assets. It's immutable, preventing alterations or deletions post-recording. It enables direct peer-to-peer transactions, eliminating intermediaries, and undergoes validation by a node network for data integrity [15], [16]. In our research, we explore blockchain as a decentralized registry, authorization, and authentication server, providing immutable storage. Yuki et al. [17] proposed a system using blockchain for authentication and authorization data storage. Blockchain technology comprises permissioned (private) or permissionless (public) network systems. The permissionless system allows any user to access the system by solving a challenge. In the realm of permissioned blockchains, the HLF framework is a popular choice. Within this context, the second facet is blockchain technology, which is crucial as an authentication and authorization server. Distributed ledger technology is used to verify a user's digital identity securely [18]. HLF is a leading product in the permissioned blockchain space. How HLF works in security, authenticating, and authorizing frameworks fits perfectly with its role in building security and trust [19].

3.1 Hyperledger Fabric

Hyperledger, initiated by the Linux Foundation in 2015, collaboratively advances cross-industry blockchain tech. HLF, a standout project, shares standard blockchain features but stands out for its private, permissioned nature. Participants enroll through a trusted Membership Service Provider (MSP), ensuring higher trust and security [20]. Fabric allows flexible user management, diverse ledger data formats, consensus mechanism switching, and supports various MSPs. It introduces channels for selective transaction confidentiality among participants [21]. Employing HLF for the authentication system enhances performance and security.

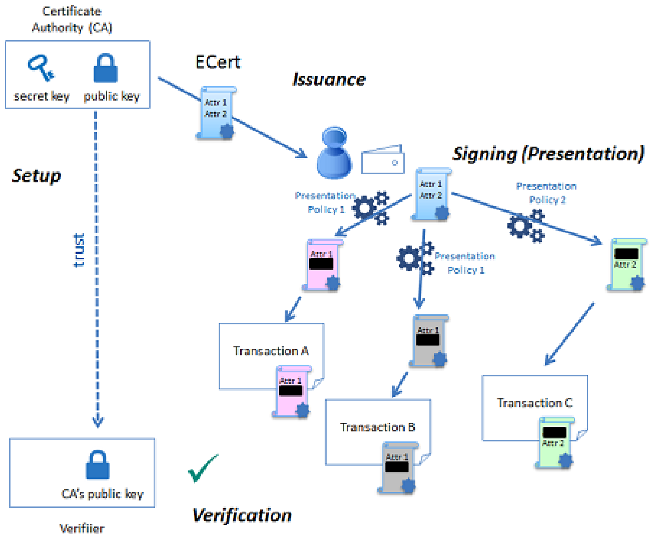


Figure 2: Identity mixer [20]

3.2 Identity Mixer

Integrated into HLF as an open-source project, It encompasses a collection of cryptographic protocols designed for authentication, signatures, and the transfer of certified attributes. Within this protocol, a series of foundational cryptographic algorithms enhance privacy by preventing linkage and reducing attribute disclosure to a minimum [20]. Figure 2 illustrates the operation of the MSP instantiated with Identity Mixer protocols. The initial step involves configuration, where the Certificate Authority (CA) signing key pair is generated, and the public key is made available to blockchain participants. Secondly, in the enrollment (issuance) phase, a peer or client generates a secret key and creates a request for an enrollment certificate (ECert). The CA issues an ECert as an Identity Mixer credential, including the member's attributes. The ECert is stored with the corresponding credential secret key on the peer side or by the client SDK. Next, the third step revolves around transaction signing (presentation). When a client (or possibly a peer) needs to sign a transaction, it generates a fresh, unlinkable presentation token that performs three functions:

1. Signing the transaction content,
2. Proving possession of a valid ECert issued by the CA,
3. Disclosing the attributes required by the access control policy for the transaction.

Finally, the last step involves verifying transaction signatures (verification), where the token is verified using the AC's public key [20]. Upon thoroughly examining the intricacies within the overarching handover process, it becomes apparent that these procedural steps encompass a multitude of vital particulars requiring improved accessibility for end users. These particulars contain elements such as identities, names, slice attributes, UE identities, and the slices in which a UE can establish connections. Consequently, to augment the comprehensive visibility and bolster the security of this dataset, the exploration and implementation of an authorized

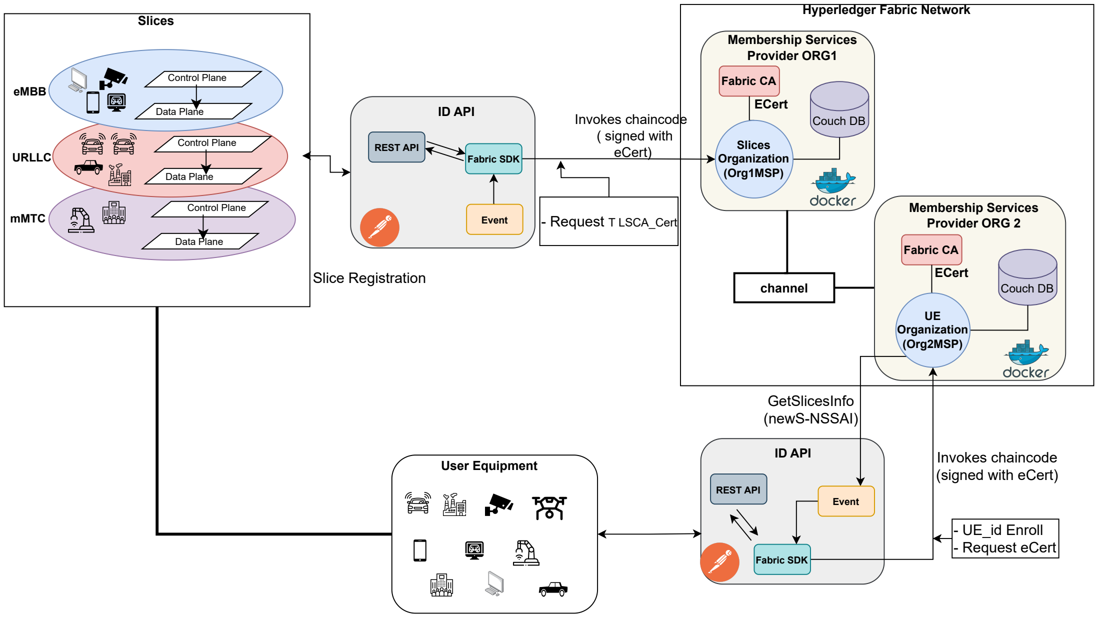


Figure 3: Proposed System

blockchain via HLF has been contemplated. Detailed insights into the undertaken efforts are expounded upon in the ensuing section.

4 Proposed System

The 5G mobile network faces numerous requirements, necessitating high flexibility and scalability in service provision. Network slicing, leveraging the capabilities of next-generation networks through SDN and NFV, emerges as a practical solution that facilitates the creation of distinct network slices, enabling efficient deployment and dynamic management of network resources and services. Moreover, HA enhances user experience by enabling seamless transitions between network slices without the need for re-authentication. Thus, it ensures uninterrupted service while upholding data exchange integrity, security, and confidentiality. This chapter delves into the detailed proposed system and its performance analysis.

4.1 System Outline

Figure 3 describes a complex system that enables users to benefit from multiple services offered by the new generation of mobile networks from different network slices. Network slicing and a HLF make this possible, enabling continuous authentication during the network handover. HLF facilitates the connection between participants. Both network participants maintain a peer-to-peer connection with the chaincode and the ledger. Each participant must obtain membership to be involved in the HLF network, and ECert are issued to participants through the CA. These certificates serve as identities for the participants, enabling the execution of transactions in the HLF environment, where chaincode procedures are invoked and signed by ECert.

4.1.1 Network architecture

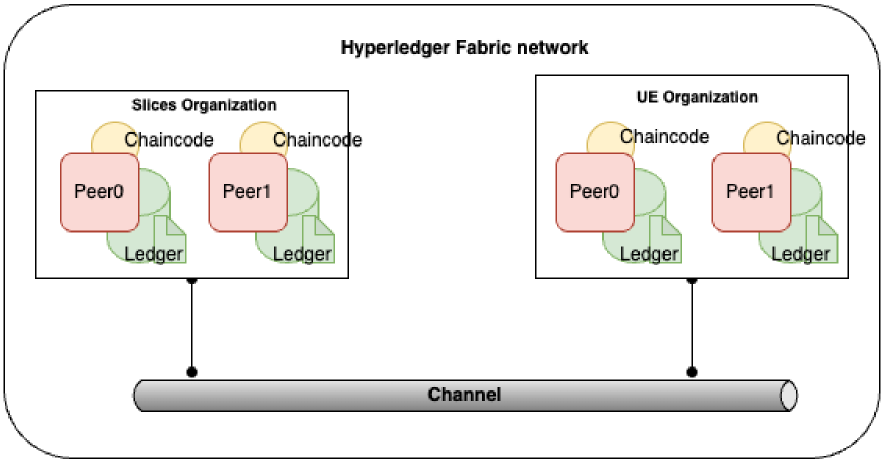


Figure 4: Network architecture

Figure 4 depicts the architecture of the HLF network designed for the HA process. Both organizations, namely the slice organization and the UE organization, are part of a unified channel. Within a HLF network, a channel is associated with a ledger, and access to this ledger is restricted solely to peers connected to that particular channel. To ensure data redundancy, each organization maintains two peers capable of internal communication, ensuring uninterrupted communication in case of peer failure. One peer from each organization assumes an anchor role to facilitate communication between organizations and streamline communication within the channel. Consequently, each organization possesses an identical copy of the ledger, containing all the necessary data for the handover and authentication process. Due to the characteristics of distributed ledger technology (DLT), this ledger ensures data integrity, as well as the immutability and security of data. In this network, both organizations adhere to the *RAFT* consensus protocol, where a cluster of two servers can continue to operate even if one of the servers fails. However, a server cluster comprises leaders, candidates, and followers. In [22], An *AppendEntry* is triggered when a client issues a command to the server, and a leader initiates this command to instruct the servers to update or synchronize their ledgers to match the new data. The modification is validated when all the servers execute it, and finally, the leader sends the execution result to the client. HLF uses the *RAFT* consensus because it offers a high success rate and high throughput.

Chaincode The chaincode allows organizations to update the stages of the HA process, as it is installed on the peers and enables them to interact with the channel. Chaincodes only permit one organization to update its necessary data for operation. As part of this process, the chaincode is designed and programmed to validate the process and only approve transactions from the correct organization. The chaincode allows users from organizations to provide data during a process. It is also worth noting that the chaincode is designed and programmed to validate a process in progress and only approves transactions sent by the correct organization during the handover process. These data contain information related to the authentication process during the handover. Therefore, at each handover, there is at least one asset in the chaincode, such as the identity of the UE and that of the target slice and their characteristics.

Consequently, the chaincode provides additional functionality to store data from the handover process, query existing data, and create and complete the history of UE states within the network.

REST API The chaincode is invoked through the Fabric SDK. Therefore, the REST API in this work plays a crucial role in facilitating communication between UEs and slices by interacting with the Fabric SDK. Postman is employed to create the REST API, aiming to streamline the integration of blockchain technology and shield the complexity of this technology from users. During the network's creation, the identities of the network's participants are generated using the key generation tool and stored in a wallet along with the corresponding certificates and keys. However, in the event of an occurrence, the Fabric SDK is used to establish a connection with the HLF network. It is also utilized to verify certificates and register users in the wallet, ensuring that only users with the correct credentials can access the network and invoke the chaincode. When a new user connects, access to the network is restricted until their identity has been recorded in the wallet using the new certificates.

In this work, blockchain technology provides high-quality and secure services in the handover process. HLF enables the blockchain network to become private, granting access and participation in the system only to stakeholders. To maintain system security, the combination of these two elements is essential for ensuring confidentiality and security within HLF:

- MSP: MSP provides and manages identities through the CA for all authenticated participants. Utilizing cryptographic mechanisms and protocols, MSP issues and verifies certificates during user authentication, as well as defining its identity policies and rules for their use (generation and signature verification) and regulations (identity validation). Each node must have a defined local MSP, and the peer must always verify the channel's MSPs before initiating the chaincode.
- Access Control: It contains a checklist (ACL) that allows limiting permissions for various network operations. For instance, it can authorize a UE to invoke the chaincode to connect only to the slice contained within its associated NSSAI and restrict its access to other slices.

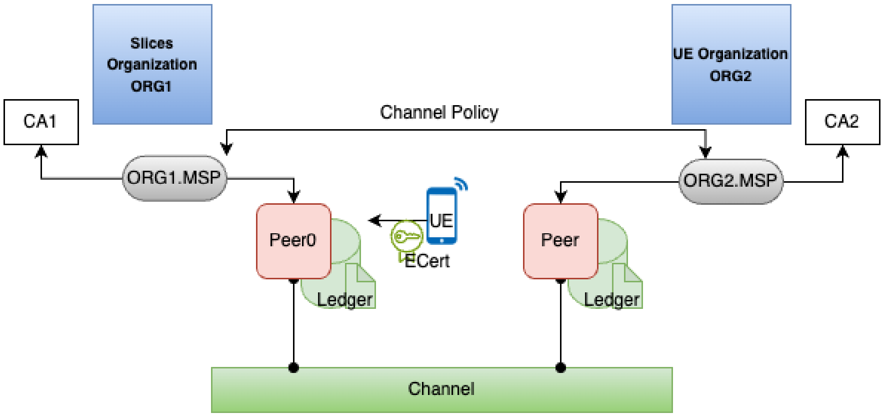


Figure 5: Handover process in MSP

Figure 5 illustrates the authentication and verification process during handover using data from two organizations: the slices organization (ORG1), which contains the slices (including their attributes and characteristics), and the UE Organization (ORG2) with UE identifiers and NSSAI associated with these identifiers. CA1 issues an identity to a UE requesting handover. Using this identity, the UE connects to the peer and attempts to invoke a codechain on the peer. The peer verifies the UE local MSP (ORG2-MSP) using the channel policy to confirm that the identity belongs to ORG2. If the verification is successful, the handover is executed, and the UE instantiates a chaincode on the channel to store its current state, with the agreement of all participants in the channel.

Algorithm 1 Enroll UE and Register Slice

```
procedure EnrollAndRegister($UE\_id, S\_NSSAI$)
    $Ecert, TLSCA\_Cert$ ← Empty Strings
    $Local\_Storage$ ← InitializeLocalDatabase()                    ▷ Initialize local storage
    $Ecert, TLSCA\_Cert$ ← EnrollUE($UE\_id$)
    $Slice\_Registration\_Request$ ← CreateSliceRegistrationRequest($S\_NSSAI$)
    $Response$ ← RegisterSlice($Slice\_Registration\_Request$)
    if $Response$ is successful then
        $Ecert\_private\_key$ ← GeneratePrivateKey()                 ▷ Generate private keys
        $TLSCert\_private\_key$ ← GeneratePrivateKey()
        $Store(Ecert, Ecert\_private\_key, TLSCert\_private\_key, TLSCA\_Cert)$
        $SaveToLocalStorage(Local\_Storage, Ecert, TLSCA\_Cert)$
        return " UE Enrollment and Slice Registration Successful"
    else
        return "Slice Registration Failed"
```

We will explore three algorithms used in the simulation of the proposed system. These algorithms constitute the fundamental foundation of the simulation, offering various perspectives and approaches for network analysis and behavior evaluation to assess the architecture's performance. The algorithm 1 presents a detailed technical process for UE enrollment and network slice registration, all while meticulously managing certificates and private keys. Initially, it initializes `Ecert` as empty strings intended to store certificate information. Additionally, it sets up a local storage database. Subsequently, the algorithm calls the `EnrollUE` function, utilizing the `UE_id` as an input to obtain essential certificates for the UE. A network slice registration request, denoted as `Slice_Registration_Request`, is created, incorporating the provided `S_NSSAI`. The algorithm proceeds to attempt the registration of the slice, sending the registration request to the network and scrutinizing the response, `Response`, to verify its success. In case of a successful registration, the algorithm generates private keys (`Ecert_private_key` and `TLSCert_private_key`) for the obtained certificates. To ensure secure storage, it employs the `Store` function to manage these certificates and private keys, saving this critical data to a local storage database. Ultimately, the algorithm concludes by returning "UE Enrollment and Slice Registration Successful" if the registration was successful or "Slice Registration Failed" if unsuccessful.

This algorithm is a foundational framework for enrolling UEs and registering network slices, guaranteeing the secure handling of certificates and private keys—an essential component in establishing and managing network services, especially in contexts necessitating robust security measures within HLF Networks.

The `ObtainSlicesInfo`, algorithm 2 takes a UE identifier, represented as `UE_id`, as input. It serves the purpose of retrieving essential information related to network slices and the TLS-CA Cert. Within the algorithm, the `GetSlicesInfo` function is invoked with the `UE_id`

Algorithm 2 ObtainSlicesInfo

```
procedure ObtainSlicesInfo($UE\_id$)
    $S\_NSSAI$ ← GetSlicesInfo($UE\_id$)                ▷ Retrieve Slice Information
    $TLS\_CA\_Cert$ ← GetTLSCACertificate()           ▷ Retrieve TLS-CA Certificate
    return $UE\_id, S\_NSSAI, TLS\_CA\_Cert$
```

to acquire the corresponding S-NSSAI. The algorithm retrieves the TLS-CA certificate using the `GetTLSCACertificate` function. Ultimately, the algorithm returns three critical pieces of information as output: the original `UE_id`, the obtained S-NSSAI, and the TLS-CA certificate encapsulating a fundamental process for obtaining crucial data within a HLF Network.

Algorithm 3 EcertAuthentication

```
procedure EcertAuthentication($UE\_id, Ecert$)
    $IsEcertValid$ ← VerifyEcert($UE\_id, Ecert$)          ▷ Verify the Enrollment Certificate
    if $IsEcertValid$ then
        $S\_NSSAI$ ← GetSlicesInfo($UE\_id$)                ▷ Retrieve Slice Information
        $TLS\_CA\_Cert$ ← GetTLSCACertificate()           ▷ Retrieve TLS-CA Certificate
        $AuthStatus$ ← AuthenticateUE($UE\_id$)              ▷ Authenticate the UE
        $PolicyRules$ ← GetPolicyRules($UE\_id$)             ▷ Retrieve Policy Rules for UE
        $TrafficStats$ ← CollectTrafficStats($UE\_id$)       ▷ Collect UE Traffic Statistics
        return $UE\_id, S\_NSSAI, TLS\_CA\_Cert, AuthStatus, PolicyRules, TrafficStats$
    else
        return ”Ecert Verification Failed”                  ▷ Ecert verification failed
```

The algorithm 3 is designed to verify the Ecert associated with a UE and, if valid, facilitate network access for the UE within a network slicing context. It begins by verifying the Ecert's validity through the ”VerifyEcert” function. If the Ecert is valid, the algorithm proceeds to retrieve critical information for network access, including Slice Information (S_NSSAI), the TLS-CA_Cert, the authentication status of the UE (AuthStatus), policy rules governing the UE's behavior, and traffic statistics (TrafficStats):

- AuthStatus: Authentication is a critical step in network access control. The ”AuthStatus” indicates whether the UE attempting to access the network slicing has been authenticated. Authentication typically involves verifying the identity and authorization of the UE. This step ensures that only legitimate devices can connect to the network slice, preventing unauthorized access and potential security breaches. For example, it might involve the UE providing credentials such as a username and password or using a secure authentication protocol like EAP.

- PolicyRules: Policy rules define how the UE can use the network slice once authenticated. These rules specify Quality of Service (QoS) requirements, traffic prioritization, and usage limits. For instance, a policy rule might ensure that a UE receives a minimum guaranteed bandwidth for a specific application or restricts the UE's access during certain hours to manage network congestion. These policies are essential for controlling network resources efficiently, meeting service level agreements (SLAs), and providing a consistent user experience.

- TrafficStats: Traffic statistics refer to the data collected about the UE's network usage.

This includes metrics like the amount of data transferred, the transfer rate, latency, and packet loss. Network operators can gain insights into how UEs utilize network resources by monitoring traffic statistics. For example, they can identify which applications or services generate the most traffic and ensure that QoS parameters set by policy rules are met. This data is valuable for optimizing network performance, troubleshooting issues, and making informed decisions about network capacity planning.

This comprehensive approach ensures that authenticated UEs gain access to network slicing with the appropriate settings, security, and policies. If the Ecert verification fails, the algorithm returns an indication of the failure, preventing unauthorized access attempts. Examining these three algorithms in the context of network slicing and HLF networks highlights their essential role in ensuring network services' secure and efficient operation. The registration, verification, and access control in the Fabric network guarantee that authenticated UE gains access to network slices with the appropriate security and policy settings, thereby contributing to effective resource allocation and a seamless user experience.

4.1.2 Workflow

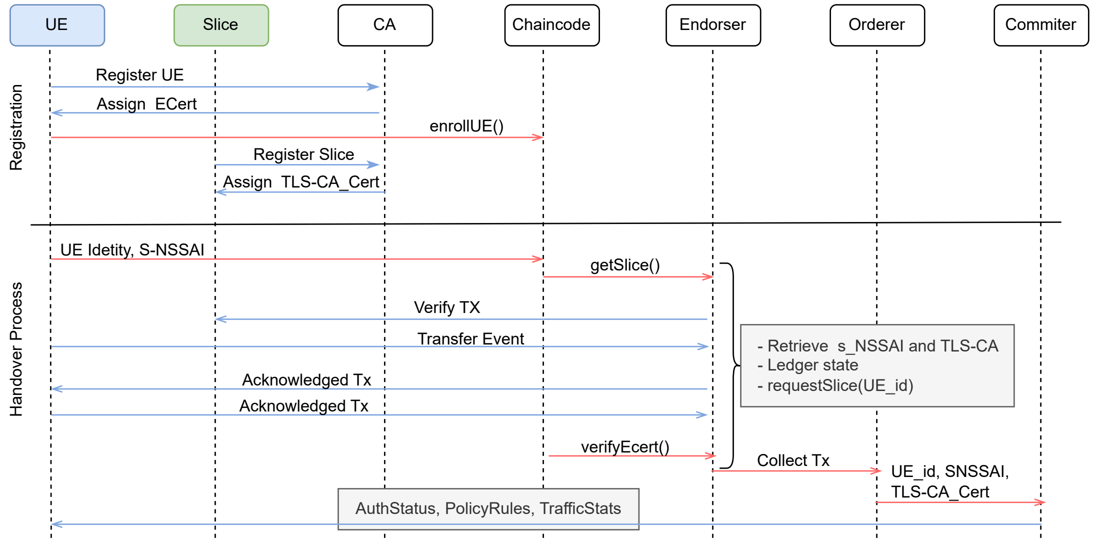


Figure 6: Workflow within HLF network

The sequence diagram, illustrated in Figure 6, visually depicts the progression of messages and interactions among diverse components within a transaction, offering valuable insights into the system's behavior and the chronological sequence of operations during the transactional process. The workflow encompasses key actions, including the UE and slice registration, assignment of certificates, verification of transactions, and the collection of transactional data. Notably, the presence of the "Slice" element and various associated functions suggests a dedicated emphasis on managing slices within the framework of the proposed system.

4.2 Performance Analysis

Simulations are performed using a computer with an Intel Core i7-10700, 2.90GHz, 16 Core processor, and 32 GB memory running Ubuntu 20.04.01. The development environment comprises HLF 1.4.7, Docker 24.0.5, Docker-composer 1.29.2, Couch-DB v1.4, and Go for code development. For consensus, RAFT is used, and caliper is used for network analysis. Caliper is employed to conduct performance tests on this system, thereby assessing the performance of blockchain network implementations and capturing relevant metrics, including transactions per second (TPS), transaction latency, transaction throughput, etc. It is also utilized for individual blockchain analysis to evaluate their suitability for the handover process within UE, slices, and handover. For the scope of this implementation, a Caliper performance test was employed to generate a report focusing on transaction latency and throughput. Transaction latency t_L signifies the time required for transactions within the network. In contrast, transaction throughput t_T is calculated based on transaction confirmation time and the number of successful transactions executed on UE.

$$t_L = (c_t \times n_t) - s_t$$

With c_t: transaction confirmation time, n_t: network, and s_t: submit time for the transaction in the HLF network.

And

$$t_T = \frac{t_{ct}}{t_{ts}} \times n_{ue}$$

With t_{ct}: transaction committed on the entire network, t_{ts}: time taken to execute transactions successfully on n_{ue}, and n_{ue}: number of UE in the network on which transactions are committed.

Table 1 presents a comprehensive report on the impact of varying transaction quantities and throughput on the performance of the HLF network. The automated test considered two critical transaction types: slice request and transfer processes. These transactions involve read and write operations on the blockchain network, all executed at a fixed rate of 1700 TPS (transactions per second). The collected data, summarized in the table above, describe how network performance evolves with increasing users (10, 20, 30, 40, 50). Metrics such as average transaction time, average request response time (t_L), and average throughput (t_T) allow us to draw crucial conclusions about the scalability, stability, and efficiency of the HLF system in network slicing handover scenarios. This data is essential for assessing the network's capacity to meet the European Union's needs and provides vital insights for decision-making regarding the security and optimization of the 5G or 6G network.

Table 1: Performance Metrics

Users	**Avg. Transaction Time (s)**	t_L **(s)**	t_T **(Mb/s)**
10	1.097	0.086	1.467
20	1.312	0.101	2.915
30	1.739	0.127	3.733
40	1.448	0.125	4.413
50	1.195	0.090	4.850

Figure 7 presents the graph representing a proposed system's performance based on user load. More precisely, it illustrates how three key metrics evolve as the number of users increases.

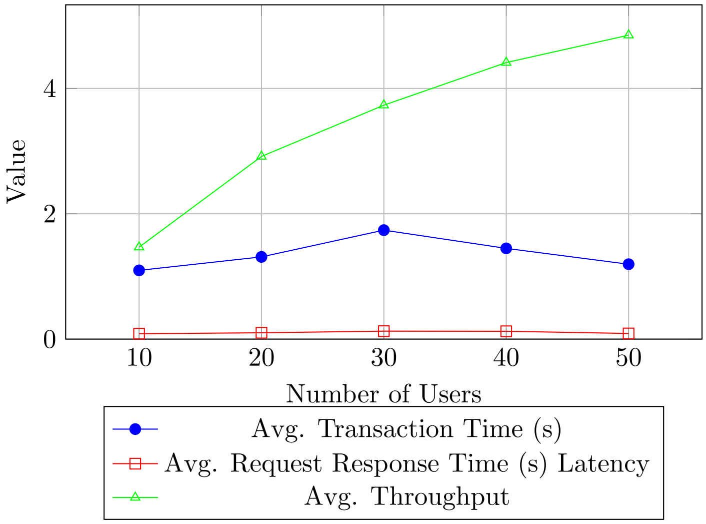


Figure 7: Performance Metrics for handover process

The average transaction time shows that despite increasing users, transaction time remains relatively stable, indicating consistent system performance. Furthermore, the average request response time also maintains remarkable stability despite the increase in load. Lastly, the system's efficiency, measured by the average throughput, significantly increases with the number of users. This suggests that HLF enables scalable management of many users while maintaining consistent performance, offering a distinct advantage in scalability and efficiency compared to other blockchain solutions.

4.2.1 Additional Computation

Table 2: Computation Overhead for Registration per UE

UE	Throughput (Mbps)	Tx/s	Latency /ms
10	5.506	0.0546	0.825
20	5.595	0.05095	0.80305
30	4.673	0.0443	0.69277
40	4.32	0.03776	0.60843
50	3.54	0.03286	0.53288

Table 2 explores the computational overhead during the registration process for each UE. The overhead is presented in terms of Throughput Overhead, Tx/s Overhead, Latency Overhead/ms. The values in this table represent the additional computational load imposed on the system when a UE engages in the registration process. As the number of UEs increases, the overhead values provide insights into the extra computational burden on the system. This table allows for an analysis of the system's efficiency and capacity during the registration phase, helping to understand how the computational load scales with the number of UEs. On the other hand, Table 3 provides indicators related to the HA process for different numbers of UE. The indicators include Total Throughput (Mbps), Total Tx/s (transactions per second), Laten-

Table 3: HA Process Computation

UE	Total Throughput (Mbps)	Total Tx/s	Latency/ms	UEs Iterations
10	55.06	0.546	8.25	39
20	111.9	1.019	16.061	65
30	140.2	1.329	20.783	84
40	172.8	1.506	24.289	93
50	182.7	1.662	26.644	101

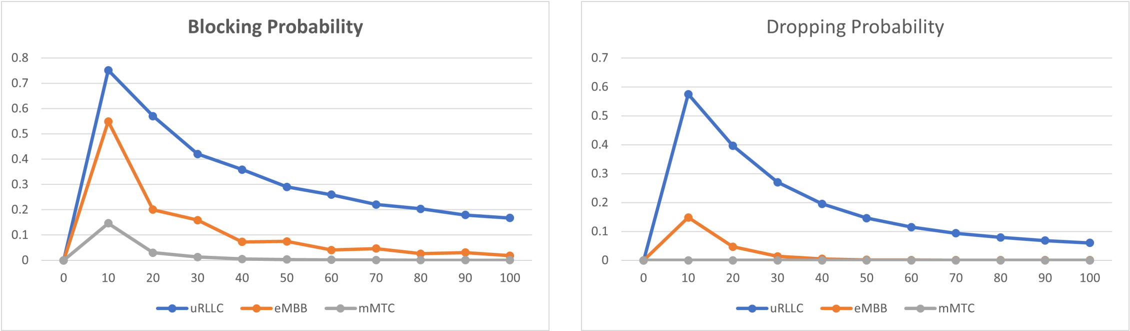
cy/ms, and UEs Iterations. These values represent the performance metrics of the HA process, showcasing how the system handles handover authentication under varying UE loads. As the number of UEs increases, the throughput, transaction rate, and request rate also increase, indicating the system's scalability and ability to handle authentication for a growing number of UEs.

In short, Table 2 and Table 3 offer a comprehensive view of the HA process and its associated computational overhead under varying UE scenarios. These data highlight how the HA-based infrastructure's performance using HLF affects the number of EUs. A continuous increase in total throughput, TPS, and latency per millisecond is observed as the number of EUs grows. This trend demonstrates the HA architecture's capability to deliver scalable performance, a critical asset for blockchain applications. Additionally, the rising number of authentication iterations reflects the increased utilization of security mechanisms in larger-scale environments.

4.2.2 System Evaluation for Different Handover Scenarios

In order to investigate our proposed framework in various handover scenarios, we evaluate by assessing blocking and dropping probabilities through implementing a Call Admission Control (CAC) algorithm proposed in [23] that prioritizes handover calls. In this context, blocking probability refers to a connection attempt being blocked or denied due to congestion or other network-related issues, while dropping probability signifies the probability that a user will experience a dropped connection due to congestion or other network-related issues (see the detailed in [23]). In our scenario, the service provider's management of S-NSSAI involves considering the number of UEs and their slices. In this sense, each slice with its capacity and connected UEs is analyzed for handover permission. The simulation contains three slices: enhanced Mobile Broadband (eMBB), massive Machine Type Communications (mMTC), and Ultra Reliable Low Latency Communications (uRLLC), all contributing to the overall system capacity for 100 UE iterations in different slices.

Figure 8 visualizes the blocking and dropping probabilities during the HA procedure in three slices, illustrating improved system performance with increasing capacity. The observed null-dropping probability for mMTC in the 5G context may be attributed to the unique characteristics of mMTC traffic and the assumptions made in the simulation. Typically, mMTC involves a large number of devices with sporadic and infrequent communication needs, such as sensors or IoT devices. Due to the intermittent nature of their communication, these devices are less likely to experience call drops compared to other communication types like eMBB or uRLLC.



(a) Blocking Probability

(b) Dropping Probability

Figure 8: Handover Scenarios

5 Conclusion

In this paper, we aim to enhance the security of authentication processes during transfers within network slicing systems by leveraging blockchain technology. By creating a decentralized and secure registry for recording transfer events, the system simplifies the verification of device identities, thereby improving the efficiency of authentication during transfers. The proposed three-component model, encompassing network slicing, user environments, and a dedicated HLF blockchain for authentication and authorization, exploits the advanced privacy and security features of HLF. These features minimize delays, ensure data confidentiality, and offer flexibility and scalability, all contributing to an enhanced user experience during transfers. Additionally, the HLF network performance analysis, as illustrated in Table 1 and Figure 7, demonstrates that the system exhibits remarkable scalability while maintaining acceptable performance as the user load increases. These findings reaffirm the robustness of the underlying HLF-based system, which can efficiently manage evolving network scenarios while upholding service quality and network security. In summary, this work fortifies network security. It elevates service quality, especially within network slicing and authentication during transfers, by adopting a privacy and security solution based on HLF. As regulations and standardizations regarding blockchain for authentication during transfers within network slicing continue to evolve, this integration into existing network infrastructure promises to effectively meet the growing demand for network slicing and transfers.

Acknowledgement: This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the Special R&D Zone Development Project(R&D) - Development of R&D Innovation Valley support program(2023-DD-RD-0152) supervised by the Innovation Foundation, and also supported by the Republic of Korea's MSIT (Ministry of Science and ICT), under the 2022 technology commercialization capability enhancement project(2022-BS-RD-0034) supervised by the INNOPOLIS.

References

[1] A Hussein, Louma Chadad, Nareg Adalian, Ali Chehab, Imad H Elhajj, and Ayman Kayssi. Software-defined networking (sdn): The security review. *Journal of Cyber Security Technology*, 4(1):1–66, 2020.

[2] Farouk Messaoudi, Philippe Bertin, and Adlen Ksentini. Towards the quest for 5g network slicing. In *2020 IEEE 17th Annual Consumer Communications Networking Conference (CCNC)*, pages 1–7, 2020.

[3] Jose Ordonez-Lucena, Pablo Ameigeiras, Diego Lopez, Juan J. Ramos-Munoz, Javier Lorca, and Jesus Folgueira. Network slicing for 5g with sdn/nfv: Concepts, architectures, and challenges. *IEEE Communications Magazine*, 55(5):80–87, 2017.

[4] Ibrahim Afolabi, Tarik Taleb, Konstantinos Samdanis, Adlen Ksentini, and Hannu Flinck. Network slicing and softwarization: A survey on principles, enabling technologies, and solutions. *IEEE Communications Surveys & Tutorials*, 20(3):2429–2453, 2018.

[5] Shunliang Zhang. An overview of network slicing for 5g. *IEEE Wireless Communications*, 26(3):111–117, 2019.

[6] Álvaro Gabilondo, Zaloa Fernández, Roberto Viola, Ángel Martín, Mikel Zorrilla, Pablo Angueira, and Jon Montalbán. Traffic classification for network slicing in mobile networks. *Electronics*, 11(7):1097, 2022.

[7] Chun-I Fan, Yu-Tse Shih, Jheng-Jia Huang, and Wan-Ru Chiu. Cross-network-slice authentication scheme for the 5 th generation mobile communication system. *IEEE Transactions on Network and Service Management*, 18(1):701–712, 2021.

[8] Ran Liu, Xingyuan Hai, Siwei Du, Lingkang Zeng, Jie Bai, and Junyu Liu. Application of 5g network slicing technology in smart grid. In *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pages 740–743. IEEE, 2021.

[9] Jaspreet Singh Walia, Heikki Hämmäinen, Kalevi Kilkki, and Seppo Yrjölä. 5g network slicing strategies for a smart factory. *Computers in Industry*, 111:108–120, 2019.

[10] Satish Kumar, Rahul Banerji, Naman Gupta, Suman Kumar, Sukhdeep Singh, Avinash Bhat, Seungil Yoon, and Shatarupa Dash. Mas5g: Move around smartly in 5g. In *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*, pages 214–221, 2019.

[11] Abbas Yazdinejad, Reza M. Parizi, Ali Dehghantanha, and Kim-Kwang Raymond Choo. Blockchain-enabled authentication handover with efficient privacy protection in sdn-based 5g networks. *IEEE Transactions on Network Science and Engineering*, 8(2):1120–1132, 2021.

[12] Haijun Zhang, Na Liu, Xiaoli Chu, Keping Long, Abdol-Hamid Aghvami, and Victor C. M. Leung. Network slicing based 5g and future mobile networks: Mobility, resource management, and challenges. *IEEE Communications Magazine*, 55(8):138–145, 2017.

[13] Sunisa Kunarak and Raungrong Suleesathira. Multi-criteria vertical handoff decision algorithm for overlaid heterogeneous mobile ip networks. *Journal of the Franklin Institute*, 357(10):6321–6351, 2020.

[14] Vincent Nyangaresi, Silvance Abeka, and Anthony Rodgrigues. Security evaluation of cellular networks handover techniques. *International Journal of Computer Network and Information Security*, 10:45–59, 05 2018.

[15] Christian Esposito, Massimo Ficco, and Brij Bhooshan Gupta. Blockchain-based authentication and authorization for smart city applications. *Information Processing Management*, 58(2):102468, 2021.

[16] Muhammad Firdaus, Siwan Noh, Zhuohao Qian, and Kyung-Hyune Rhee. Bpfl: Blockchain-enabled distributed edge cluster for personalized federated learning. In *International Conference on Computer Science and its Applications and the International Conference on Ubiquitous Information Technologies and Applications*, pages 431–437. Springer, 2022.

[17] Yuki Ezawa, Makoto Takita, Yoshiaki Shiraishi, Shohei Kakei, Masanori Hirotomo, Youji Fukuta, Masami Mohri, and Masakatu Morii. Designing authentication and authorization system with blockchain. In *2019 14th Asia Joint Conference on Information Security (AsiaJCIS)*, pages 111–118, 2019.

[18] Muhammad Firdaus, Harashta Tatimma Larasati, and Kyung-Hyune Rhee. A blockchain-assisted

distributed edge intelligence for privacy-preserving vehicular networks. *Computers, Materials & Continua*, 76(3), 2023.

[19] Mukesh Thakur et al. Authentication, authorization and accounting with ethereum blockchain. 2017.

[20] Hyperledger fabric. https://hyperledger-fabric.readthedocs.io/en/release-2.5/blockchain.html. Accessed: 2023-09-18.

[21] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolić, Sharon Weed Cocco, and Jason Yellick. Hyperledger fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, EuroSys '18, New York, NY, USA, 2018. Association for Computing Machinery.

[22] Deebthik Ravi, Sashank Ramachandran, Raahul Vignesh, Vinod Ramesh Falmari, and M Brindha. Privacy preserving transparent supply chain management through hyperledger fabric. *Blockchain: Research and Applications*, 3(2):100072, 2022.

[23] Azhagu Jaisudhan Pazhani. A, P Gunasekaran, Vimal Shanmuganathan, Sangsoon Lim, Kaliappan Madasamy, Rajesh Manoharan, and Amit Verma. Peer–peer communication using novel slice handover algorithm for 5g wireless networks. *Journal of Sensor and Actuator Networks*, 11(4):82, 2022.