Computer Communications 200 (2023) 1–16

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Mobile-Chain: Secure blockchain based decentralized authentication system for global roaming in mobility networks

Indushree M. [a], Manish Raj [a], Vipul Kumar Mishra [a], Shashidhara R. [b], Ashok Kumar Das [c,*], Vivekananda Bhat K. [d,**]

[a] School of Computer Engineering and Technology, Bennett University, Greater Noida 201 310, India
[b] Blockchain R&D, Wipro Technologies, Bengaluru, Karnataka, India
[c] Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India
[d] Department of Computer Science and Engineering and Centre for Cryptography, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India

ARTICLE INFO

Keywords:
Mutual authentication
Blockchain
Smart-contracts
Mobility network
Security

ABSTRACT

Designing a secure and efficient authentication protocol is crucial and challenging in the mobility network. Due to the seamless roaming of mobile users over multiple foreign agents and the broadcast nature of the communication channel, the mobile networks are often exposed to several network attacks. To achieve perfect authentication and secure communication among mobility entities like MU (Mobile User), FA (Foreign Agent) and HA (Home Agent), the researchers have proposed numerous authentication protocols in the past. However, the existing protocols for the mobility environments are insufficient to address the fundamental security concerns and an adversary can impersonate the mobile user at anytime. Thus, we propose Mobile-Chain, a secure blockchain-based authentication system for mobility environments. The proposed Mobile-Chain is designed to protect user privacy and guarantees provable security like authentication, anonymity, untraceability, confidentiality, data integrity, and decentralization. The implementation of the security framework has been done on the ethereum blockchain platform using smart contracts written in a solidity programming language. The security analysis reveals that Mobile-Chain is robust against various security threats to which mobility networks are vulnerable. Besides, the authentication framework has been measured through a formal security verification tool known as Automated Validation of Internet Security Protocol and Application (AVISPA). Notably, the performance evaluation of the proposed protocol proves that it maintains performance gain, computationally efficient, and implementable in resource-limited wireless and mobility environments.

1. Introduction

With the tremendous growth of cryptocurrencies like bitcoin, blockchain technology enables decentralized, distributed, and peer-to-peer networking through consensus protocols without any intermediaries. Blockchain guarantees decentralization, transparency, and security features like authentication and integrity. When a new block is added to a blockchain, the previous block is linked to it using a cryptographic hash value. This characteristic guarantees that the chain connecting the blocks will never be broken and the network transactions are immutable [1]. Blockchain has been recognized as a promising technology for upcoming mobile networks, which allows the mobile nodes to travel to any location in the world and access ubiquitous services [2]. However, it is generally recognized that mobile environments are open to a number of security threats. The confidential data that has been transmitted over the radio link can be intercepted, blocked, and altered by an attacker. Mutual authentication and confidentiality services between the communication parties are therefore extremely necessary.

In the roaming scenario, the MU, HA, and FA make up the common authentication mechanism. After successfully registering with the Home Network (HN), the mobile user can utilize the HA's services. A Foreign Network (FN) administered by the FA is visited by the registered user [3]. Mutual authentication between the entities is essential in this case to prevent the adversaries from gaining unauthorized access. Additionally, the current authentication protocols expose highly sensitive user privacy, such as anonymity and location information. In order to prevent numerous vulnerabilities in the mobility networks, a secure and robust blockchain-based user AUC (authentication) solution is essential. The blockchain-based authentication framework is

* Corresponding author.
** Corresponding author.
E-mail addresses: indushree.june1@gmail.com (Indushree M.), manish.raj@bennett.edu.in (M. Raj), vipul.mishra@bennett.edu.in (V.K. Mishra), eemailshashi@gmail.com (Shashidhara R.), iitkgp.akdas@gmail.com, ashok.das@iiit.ac.in (A.K. Das), kv.bhat@manipal.edu (Vivekananda Bhat K.).

https://doi.org/10.1016/j.comcom.2022.12.026
Received 4 August 2022; Received in revised form 11 October 2022; Accepted 28 December 2022
Available online 31 December 2022

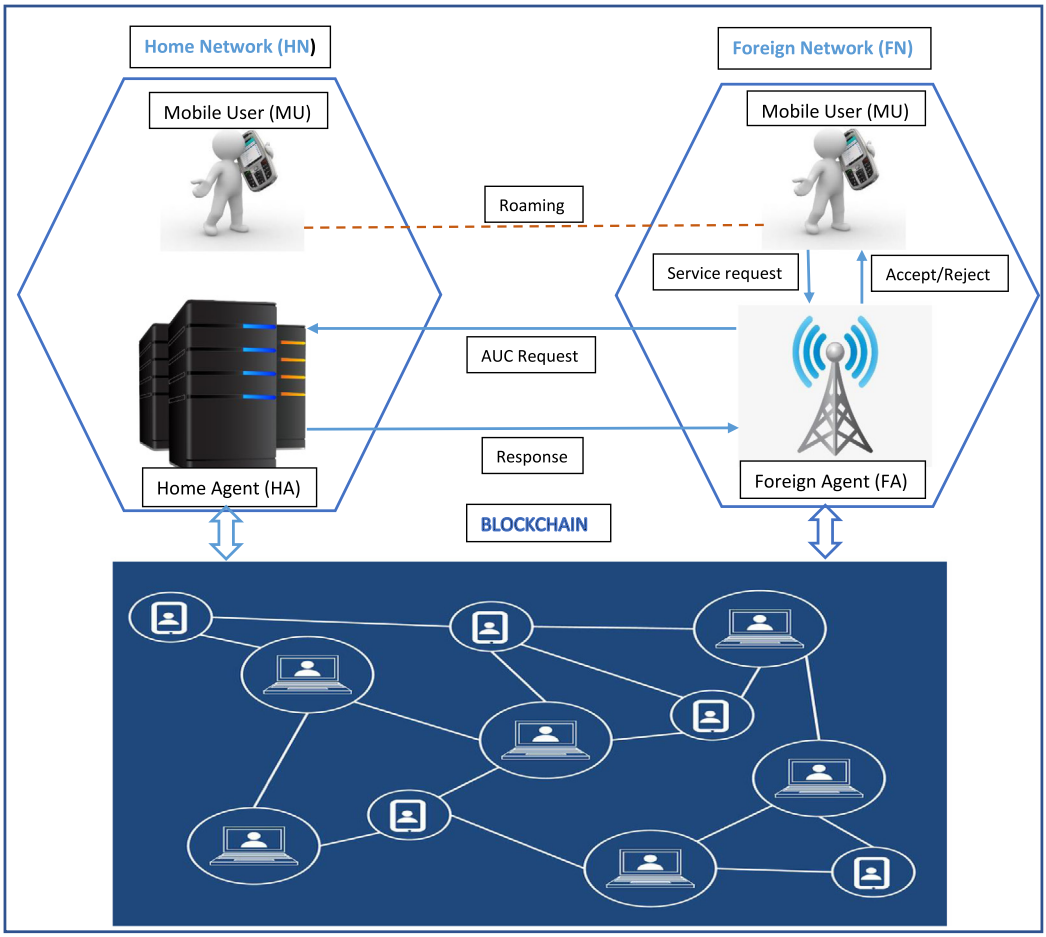

Fig. 1. Mobile-Chain: Blockchain based mutual authentication for roaming service.

proposed to afford roaming services in the mobility environment as depicted in Fig. 1.

1.1. Motivation

Presently, blockchain is clinched outstanding recognition from research organizations and academia where researchers make use of this technology for building secure frameworks. The blockchain is a digital public ledger that uses decentralized nodes, encryption technologies, smart contracts, and consensus protocols to guarantee security, transparency, and decentralization across a range of applications. The cryptographers and the industry have identified some critical issues associated with authentication and authorization in the existing roaming scenarios, which include the following:

- In the global mobility networks, the majority of the authentication protocols now in use are susceptible to well-known attacks [4]. In order to resist replay attacks, the authentication methods use a timestamp mechanism. However, the network's unpredictable delays cause the timestamp system may not work always [5].
- Two factor authentication is used by the current mutual authentication protocols. Passwords and smart cards are used for mobile user authentication. Password guessing and stolen-verifier attacks can be used against these protocols [6].
- Unauthorized access to content pertaining to mobile users was left unattended, which is thought to be a serious security violation of the user.
- All mobile users must authenticate with foreign agents through the home network before they may access the required services [7]. Unauthorized users and adversaries are prevented from gaining access to the mobile network, nevertheless.
- Additionally, the centralized infrastructures of the current authentication systems for mobile networks create significant problems during system failures or network breakdowns.

As a result, the roaming service within the framework of mobility networks requires a strong blockchain-based security architecture to assure security against various attacks.

1.2. Design goals and security requirements

The important design metrics and prime security requirements for roaming service in the mobile networks are as follows:

- *User Anonymity:* Traditional smart-card based authentication protocols do not offer privacy for mobile nodes since a malicious or compromised attacker can disclose the user's identity during the authentication phase. Therefore, the mobile user's identifying information should be kept private [8].
- *Untraceability and Unlinkability:* The intercepted messages exchanged during the authentication phase should not be linked, allowing the attacker to trace the origin or location of the mobile user [9]. Blockchain is pseudonymous in nature, therefore for the majority of its architecture, an intruder cannot determine the identities of mobile nodes by examining transactions.
- *Mutual authentication:* The protocol should come up with a secure mutual authentication phase, where MU, FA, and HA mutually-authenticate each other to resist impersonation attacks in mobile networks [10].

- *Perfect-forward secrecy:* A perfect forward secrecy technique should be used in the authentication protocol design to provide secrecy for previously sent messages, preventing an opponent from determining a past session key by learning the entity's private and public keys [6].
- *Single registration:* A single registration with the HA would be necessary for the authentication process before the mobile users could log in. Mobile user registration just needs to be done once to make it practical.
- *Session key fairness:* The authentication protocol should provide secure session key establishment, where all communication entities, such as MU, FA, and HA, participate in the session key negotiation in order to achieve secure communication [11].
- *No verifier tables:* The protocol should eliminate the need for verifier tables at HA and FA, which have been employed in conventional systems, in order to resist against stolen verifiers, insider attacks, and dictionary attacks.
- *Modification and interception resilience:* The confidentiality and integrity of the information should be protected during transmission.
- *Resistance to other attacks:* The authentication system should resist against impersonation attacks, insider attacks, replay attacks, denial of service (DoS) attacks, smart-card loss, bit-flipping attacks, and so on [12].
- *Security infrastructure:* Technology like blockchain is required because, unlike centralized systems, it prevents all information from being kept in one location. The central server failure or single point of failure is therefore nonexistent. Additionally, blockchain maintains data in distributed ledgers that guarantee data integrity, secrecy, and transparency.
- *Computational gain:* An authentication system could be effective and lightweight to address the mobile terminals' resource limitations. By including important cryptographic operations in the protocol architecture, computational efficiency can be gained.

1.3. Research contributions

The research contributions of the paper are summarized as follows:

(1) For the mobility networks leveraging blockchain technology, a novel Mobile-Chain that is a reliable authentication technique has been presented. This decentralized security architecture guarantees mutual authentication, anonymity, and resilience to diverse threats.
(2) Blockchain benefits the proposed mutual authentication system, which is secure from bit-flipping, stolen verifier, SQL injection, and dictionary attacks.
(3) Additionally, the proposed system uses a consensus mechanism to protect against node failures in mobility networks. Currently, mutual authentication between MU, FA, and HA during the roaming process in global mobility networks cannot be provided by the existing blockchain-based authentication protocols.
(4) The mobile user is connected to a secure crypto wallet in the proposed authentication framework in order to keep the authentication information provided by the HA. Because of lost or stolen smart cards, the blockchain-based protocol protects users from password-guessing attacks.
(5) To demonstrate the reliability of the blockchain-based authentication system's security, a thorough functional requirements analysis and comparison have been conducted. The protocol specification language has also been used to do a formal security validation.
(6) Implementation of the proposed framework has been done on the Ethereum blockchain platform using smart contracts written in the solidity language, which enhances security, transparency, and decentralization in the mobile network.
(7) Notably, the performance evaluation proves that the security framework possesses less communication and computational overhead as compared to the present mutual frameworks meant for mobile and cellular environments.

1.4. Structure of the paper

The structure of the research paper as follows. Section 2 presents the literature review of the authentication protocols, crypto notations, and a threat model. Section 3 presents a novel blockchain-based authentication system for the global mobility environments, and the security requirement analysis has been described in Section 4. The blockchain implementation of the protocol has been presented in Section 5. Subsequently, Section 6 demonstrates AVISPA formal verification and the corresponding results. Section 7 summarizes the security requirements and the performance analysis. Finally, the article is concluded in Section 8.

2. Background

The cryptographers, researchers, and wireless-based organizations have been aspiring in designing security various security protocols and frameworks using strong cryptographic techniques to resist various attacks in mobile environments. However, most of the conventional authentication protocols (two-factor authentication protocols) for wireless and mobility networks in the literature are prone to well-known attacks such as insider attack, SQL injection, impersonation attacks, replay attacks, and many more.

In 2011, He et al. [8] proposed a secure user authentication protocol with smart cards for wireless environments. The authors believed that their protocol provides user anonymity and prevents impersonation and replay attacks. The authors in [13] proposed a simple authentication system with anonymity for battery-powered mobile devices using wireless and mobile communications, this authentication technique can also offer effectiveness and security. However, Li et al. [14] examined He et al. [8] authentication scheme and concluded that their protocol is not user-friendly and is unable to guarantee user anonymity and fairness in the key agreement. Consequently, Li et al. [14] suggested a new authentication protocol for wireless and mobile networks that ensures user anonymity.

To strengthen the security of global mobility networks, in 2013, Jiang et al. [15] presented an improved authentication mechanism. However, Wen et al. [16] demonstrated that Jiang et al.'s scheme is susceptible to replay, stolen-verifier, and denial of service attacks, and they also suggested a new technique to overcome the security vulnerabilities of the previous authentication protocols. Later, Li et al. [9] also put out a new authentication technique, claiming that it met all security requirements for the mobility network. In 2014, Zhao et al. [6] identified that the technique proposed by Mun et al. [17] was susceptible to insider, password guessing, and forgery attacks and that it was unable to provide user anonymity, user friendliness, or mutual authentication. They then proposed a new authentication protocol to fix the security issues with the current authentication protocols.

In 2015, Marimuthu and Saravanan [18] suggested a secure authentication system with user anonymity for roaming service in global mobility networks. This protocol is thought to offer numerous advantages to withstand a variety of attacks, including the capacity to protect user anonymity and untraceability. However, Madhusudhan et al. [11] proved that their authentication scheme has several security flaws. Later, they proposed a safe and lightweight authentication mechanism for roaming service in mobile networks.

Later on, numerous authentication protocols proposed to afford roaming services in global mobile networks [5,7,19–21]. However, the current authentication procedures are more computationally complex and do not meet all security needs in the mobility network [22]. The authors in [5,22] analyzed the security strength of recently proposed

mutual authentication systems presented in [7,18] and proved that the existing protocols are prone to injection, reply, and dictionary attacks.

In wireless and mobility networks, blockchain-based protocols will be more user-friendly since they guarantee robust user authentication mechanism, tamper-proof operations, and decentralized services. Some of the blockchain-based authentication protocols are as follows: The authors in [1] proposed a novel blockchain-based roaming management protocol which consists of thoroughly analyzed Proof-of-Stake (PoS) consensus mechanism and a smart-contract-enabled roaming management platform to address the problem of roaming fraud for mobile service providers. Due to its reliance on a single-factor authentication technique, the BlockRoam protocol is unable to enable mutual authentication between the mobile subscriber, foreign agent, and home agent. The authors also fail to provide smart contracts for user authentication and blockchain implementation.

The authors in [23] used a RAFT as consensus algorithm for blockchain application of roaming services for mobile network. The raft consensus validate transactions and commits a blocks on to the ledger. It can lead to significant storage savings, particularly if the transaction load is low. However, the proposed approach is to improve the system performance during roaming process and the RAFT consensus protocol fails to address mutual authentication, user anonymity requirements.

To make the roaming service in 5G networks easier, the authors in [2] suggested a new blockchain-based architecture. For seamless connectivity regardless of MNOs connected with each 5G local operators, the suggested approach provides roaming tenants with a universal account. To address the issue of mistrust amongst MNOs (Mobile Network Operators), the authors proposed a blockchain network that is permissioned and built on smart contracts [24]. The suggested architecture use smart contracts to automatically handle billing settlement without relying on trusted third-party. Additionally, there are a number of blockchain-based protocols that have been proposed in the literature. These protocols focused on authentication in smart city applications [25], the smart grid [26], decentralized identification [27], and improving privacy preservation in fog computing environments [28–30].

Table 1 presents the merits and demerits of the security protocols which are designed to afford mobile user authentication for roaming services in global mobility environments.

2.1. System models

In this section, we discuss both the network and attacker models that are used in the proposed scheme.

2.1.1. Network model

The objective of the mutual authentication framework is to provide a reliable platform for mobile entities like MU, HA, and FA to establish secure communications over an insecure public channel. In this system, we assume that the mobility network offering ubiquitous services to mobile users using radio link in the distributed architecture. In this scenario, all the mobility entities will exchange a sequence of authentication messages over an unreliable public network like the Internet. In fact, all communication parties cannot be trusted in the global mobile environments. Indeed, the more number of mobile subscribers in the cellular or mobile network will increase the risk of security and complexity. Further, the mobility network is responsible for the transmission and reception of packets and does not ensure security services like authentication, confidentiality, and integrity. Therefore, the unauthorized mobile user can intercept, eavesdrop, or modify the sensitive information communicated through the network.

2.1.2. Attacker model

We follow the Dolev–Yao threat model [33] to describe the potentiality of the intruder and common privacy issues encountered during mutual authentication in mobility environments. The Dolev–Yao model is a very powerful adversarial model that is generally considered as the benchmark for assessing authentication protocols. An adversary in the Dolev–Yao model is capable of obtaining any message that is transmitted via the network. In order to send messages to any entity, the attacker can pose as another entity or act as a normal network user so, the Dolev–Yao model is an easy-to-use framework for examining security protocols that are frequently used in distributed systems and networks.

In addition, we also follow the honest-but-curious (HBC) adversary model [34] where a legitimate participant in a communication protocol who will not deviate from the defined protocol, but will attempt to learn all possible information from legitimately received messages.

Assume that an adversary, say $\mathscr{A}$, has the full control over a public network, where $\mathscr{A}$ could selectively drop, replay, intercept, delay, and eavesdrop sensitive information communicated between MU, HA and, FA with negligible delay [33]. In the case of the verifier or password tables in the database, the intruder $\mathscr{A}$ can easily access the information through injection attacks like cross-site scripting. Consequently, the adversary $\mathscr{A}$ could trace the MU's identity and location information when the mutual authentication framework will have some constant parameters in the message for all sessions. Furthermore, the adversary $\mathscr{A}$ will attain to extract the sensitive information from the lost/stolen smart-cards [14,17].

3. Proposed blockchain-based security framework

The security framework uses PBKDF (Parallel Blockchain Key Derivation Function) as the key derivation protocol which simplifies and improves the performance of the roaming process in the mobility networks [35].

The proposed authentication protocol is implemented using the Ethereum blockchain. Currently, the Ethereum network uses Proof-of-Stake (PoS) as a consensus algorithm. In this network model, the base stations will act as validators to verify the transactions. The mobile user terminal associated with an Ethereum wallet (similar to MetaMask) generates a public–private key-pair for the mobile user.

Before the authentication system starts, the foreign agent (FA) and the home agent (HA) establish a common secret-key using a dynamic Diffie–Hellman key exchange mechanism. In the initial phase, the HA will act as an authentication server and distribute the genesis block to all base stations. In addition, the HA collects the public keys and acts as a key distribution center. Furthermore, the mobile user attaches to the base station and receives its genesis block. In return, the mobile user provides the timestamp, random string, and its public key to the base station. Eventually, the source base station broadcasts the MU's information to all other base stations in the network for possible authentication during the roaming process.

With respect to mutual authentication, the proposed protocol comprises of the following phases, namely (1) registration phase, (2) mutual authentication phase, and (3) password change phase. The mathematical and cryptographic notations used in the system are also outlined in Table 2.

3.1. Registration phase

In this scenario, MU selects an identity ID_M and a random nonce N_M, and submits the registration request $R_1 = h(ID_M \parallel N_M)$ to the HA via a secure communication. In this case, the identity ID_M is a random ID, so MU can freely choose the identity similarly to a username.

HA retrieves an MU request and computes

$$H_M = h(R_1 \parallel S_{HA})$$

Table 1
Merits and demerits of the existing authentication protocols in mobility environments.

Authors	Research article	Published Year	Merits	Demerits
He et al. [8]	The user authentication protocol using smart-card for mobile communication	2010	User anonymity, single registration and computational efficiencies.	Susceptible to replay attacks and clock synchronization problem.
Yoon et al. [13]	A secure mutual-authentication for roaming in mobile communications	2011	Preserve privacy, anonymity and provides reliable roaming services	Prone to insider attacks, unfairness in session-key computation.
Li et al. [14]	A novel privacy-preserving authentication using smart-card for wireless communication	2012	Fairness in key agreement, forward secrecy and resistance to impersonation attacks	No local password verification, inefficient due to computational complexity.
Niu and Li [9]	An enhanced remote user authentication protocol for roaming service	2013	Local password verification and provides perfect forward secrecy	Susceptible to insider attacks and DoS attacks.
Zhao et al. [6]	An anonymous authentication protocol for the mobile networks	2014	User anonymity, local password verification and no password tables	Susceptible to replay attacks and DoS attacks.
Sarvanan and Muttu [18]	A secure authentication protocol with anonymity in global mobile networks	2015	Provides privacy, anonymity and untraceability.	Prone to password guessing attack, stolen verifier attack and forgery attacks.
Gope and Hwang [19]	Energy efficient and light weight authentication protocol for secure communications in mobility networks	2016	Provides energy efficiency and computational gain	Vulnerable to DOS attacks and suffers from clock synchronization problems.
Lee et al. [20]	A mutual authentication protocol for roaming service in the mobile network	2017	Prevents impersonation attacks and smart-card loss attack	Prone to replay attacks and insider attack.
Xu et al. [7]	An efficient mutual authentication protocol for global mobility networks	2018	Secure against clock synchronization problem, improves performance gain.	Vulnerable to Bit-flipping and Denial of Service attacks.
Madhusudhan et al. [31]	A secure mobile user authentication with privacy-preserving for mobile networks.	2019	Secure against various attacks in the mobility networks and computational efficient	Susceptible to SQL injection and dictionary attacks.
Ahmadi et al. [21]	A secure session key agreement authentication in mobility network preserving anonymity.	2019	Attains anonymity, untraceability and mutual authentication	Vulnerable to impersonation and replay attacks.
Shashidhara et al. [10]	Anonymous mutual authentication scheme for roaming in Resource-Constrained mobile devices.	2019	Resistance to DoS attacks, replay attacks and provides local password verification	Prone to Bit flipping and dictionary attacks.
Sohail et al. [32]	An improved authentication scheme for GLOMONET	2020	Provides a secure authentication and fair session-key agreement	Vulnerable to guessing and smart-card loss attacks.
Nguyen et al. [1]	Blockchain-based roaming management system for future mobile networks	2021	To address roaming fraud in mobility networks.	Single factor authentication and fails to provide mutual authentication.
Weerasinghe [2]	Blockchain-based roaming and offload service platform for local 5G operators	2021	This approach provides roaming tenants with a universal account	Fails to provide mutual authentication between MU, FA, and HA.

Table 2
Cryptographic notations used in the paper.

Notation	Description
MU, HA, FA	Mobile user, home agent, and foreign agent, respectively
ID_M, ID_H, ID_F	Identity of MU, FA, and HA, respectively
PW_M	User password
S_{FA}	FA's shared secret-key
S_{HA}	HA's private-key
$(E/D)_K$	Encryption/decryption using the key K
C_M	Counter variable
$h(\cdot)$	Hash algorithm
SK	Session-key
$\|\|$	Concatenation
$\mathscr{A}$	An active/passive attacker
$\oplus$	Bitwise exclusive-OR

Table 3
Summary of registration phase.

Mobile user		Home agent
Choose ID_M, PW_M, N_M		
Computes: $R_1 = h(ID_M \|\| N_M)$		
	$\underline{\boldsymbol{R_1 = \{h(ID_M \|\| N_M)\}}}$	
		$H_M = h(R_1 \|\| S_{HA})$
		Initialize $C_M = 0$
		HA stores $\{R_1, C_M\}$ on Blockchain
	$\underline{\boldsymbol{R_2 = \{H_M, C_M, h(\cdot)\}}}$	
MU selects a password PW_M		
Computes: $M_P = h(ID_M \|\| PW_M \|\| N_M)$		
MU stores $\{H_M, M_P, C_M, N_M\}$ on wallet		

where S_{HA} is a secret key of HA. Subsequently, HA initializes the counter $C_M = 0$ for the mobile user and stores $\{R_1, C_M\}$ on the blockchain. Moreover, HA sends the registration response $R_2 = \{H_M, C_M, h(\cdot)\}$ to MU through secure communication.

After reception of R_2 from HA, MU chooses a password PW_M and calculates

$$M_P = h(ID_M \| PW_M \| N_M).$$

Hereafter, MU stores the credentials $\{H_M, M_P, C_M, N_M\}$ on his/her wallet to access the ubiquitous services from the mobility network. The registration phase is depicted in Table 3.

3.2. Mutual authentication and session key negotiation

In this phase, a registered MU roams into FN (Foreign Network) to access desired services. Hither, the mutual authentication happens

between an MU, FA with the assistance of the HA. In addition, the protocol uses a dynamic Diffie–Hellman key exchange mechanism to distribute a shared-key in between FA & HA. The login and authentication procedure is outlined as follows.

A1: The mobile user inputs the login credentials like identity ID_M and password PW_M. The MU device retrieves parameters from his/her wallet and computes $M_P^* = h(ID_M \| PW_M \| N_M)$, and compares $M_P^* \stackrel{?}{=} M_P$. If verification is unsuccessful, the protocol will terminate the session. Otherwise, MU device produces a random nonce R_M and calculates the following:

$$M_A = h(ID_M \parallel N_M) \oplus R_M,$$

$$M_B = h(H_M \parallel C_M) \oplus R_M.$$

Then, MU forms a service request $M_1 = \{M_A, ID_H, M_B\}$ to the FA.

A2: Upon receiving the service request M_1, FA generates a nonce R_F and finds

$$F_A = h(M_A \| S_{FA}) \oplus R_F;\ F_B = h(F_A \| S_{FA})$$

where S_{FA} is a shared-secret key of the FA. In addition, FA stores $\{R_F, M_A, M_B\}$ on the blockchain. Hereafter, FA sends the authentication request $M_2 = \{ID_F, F_A, F_B, M_B\}$ to HA.

A3: HA hears the request from FA and calculates

$$S_{FA} = h(ID_F \| S_{HA});\ F_B^* = h(F_A \| S_{FA}).$$

HA checks $F_B^* \stackrel{?}{=} F_B$. If the comparison is unsuccessful, HA terminates the authentication protocol. Otherwise, HA authenticates FA. Later, HA derives C_M, R_1 from the blockchain and calculates

$$H_M^* = h(R_1 \| S_{HA});\ R_M^* = h(H_M^* \| C_M) \oplus M_B,$$

$$M_B^* = h(H_M^* \parallel C_M) \oplus R_M^*$$

and checks $M_B^* \stackrel{?}{=} M_B$. If the comparison is unsuccessful, HA ends the authentication protocol. Otherwise, HA authenticates MU and finds the following:

$$R_F = h(M_A^* \parallel S_{FA}) \oplus F_A$$

$$H_A = h(ID_H \| M_B^* \| S_{FA})$$

$$H_B = h(H_M^* \| ID_F \| C_M)$$

$$H_C = h(ID_H \parallel R_M^*) \oplus R_F.$$

Then, the value of C_M is incremented by 1 and stores on blockchain. Finally, HA replies the authentication response $M_3 = \{H_A, H_B, H_C\}$ to FA.

A4: Upon receiving an authentication response M_3, FA retrieves $\{M_B, R_F, M_A\}$ from Blockchain and computes $H_A^* = h(ID_H \| M_B \| S_{FA})$. Next, FA checks $H_A^* \stackrel{?}{=} H_A$. If the comparison is unsuccessful, FA cancels this authentication process. Otherwise, FA ensures the mutual authentication with the HA and calculates a session key as follows:

$$SK = h(M_A \| R_F \| ID_H).$$

Further, FA forwards the authentication response $M_4 = \{H_B, H_C\}$ to MU.

A5: Upon reception of the message M_4, the user computes $H_B^* = h(H_M \| ID_F \| C_M)$ and compares $H_B^* \stackrel{?}{=} H_B$. If the comparison fails, MU refuses the connection with FA and HA. Otherwise, MU mutually authenticates FA and HA. In addition, MU derives $R_F = h(ID_H \parallel R_M) \oplus H_C$ and computes the session key $SK = h(M_A \| R_F \| ID_H)$ to access the secure ubiquitous services from FA. Finally, MU updates C_M with $C_M + 1$ and session-key details on the blockchain.

The login and mutual authentication procedures are summarized in Table 4.

3.3. Password renewal phase

Here, an authorized mobile user with valid registration parameters can change his/her default password locally. The steps in password renewal phase are as follows:

S1: A registered mobile user submits the credentials like identity ID_M and password PW_M through the user interface.

S2: The device computes $M_P^* = h(ID_M \| PW_M \| N_M)$ and compares $M_P^* \stackrel{?}{=} M_P$. If the comparison fails, the password renewal phase will be aborted. Otherwise, the local password verification succeeds and legality of the mobile user will be ensured.

S3: Then, the mobile user inputs the new password PW_M^N and device calculates:

$$M_P^N = h(ID_M \| PW_M^N \| N_M).$$

S4: Finally, the device replaces M_P with a new value of M_P^N, respectively. Finally, MU holds the parameters $\{H_M, M_P^N, C_M, N_M\}$.

4. Security analysis

In this section, a rigorous security analysis has been presented for the proposed system. The attacker $\mathscr{A}$ attempts to break the security protocol using the leaked information during the communication between MU, HA, and FA. However, the adversary $\mathscr{A}$ will be unable to gain access and control over the proposed security protocol. Eventually, the blockchain based security protocol resilience against network attacks and satisfy all the requirements in global mobility environments.

4.1. Mobile user privacy and anonymity

The proposed security protocol is implemented on blockchain, which provides the user privacy. An MU do not have to provide any personal information like username, email IDs to the blockchain during registration. The user's only deal with the public Ethereum addresses. Therefore, the protocol preserves privacy. Consequently, during MU registration with the HA, MU's identification details ID_M, N_M are compressed using SHA-256 function and the registration request R_1 will be sent to HA. In this regard, an intruder $\mathscr{A}$ is unable to get the identity information. As a result the mobile user in the roaming process is remain anonymous. Suppose an attacker intercepts messages $M_1 = \{M_A, ID_H, M_B\}$, $M_2 = \{ID_F, F_A, F_B, M_B\}$, $M_3 = \{H_A, H_B, H_C\}$, $M_4 = \{H_B, H_C\}$ transmitted between MU, HA, FA in the authentication process. However, it can be observed that the identity of the mobile user ID_M is not disclosed in any of the messages. Hence, the proposed protocol ensure the mobile user privacy and anonymity.

4.2. Mutual authentication

The proposed protocol mutually authenticates all communication entities through $\{M_1, M_2, M_3, M_4\}$. The steps in the mutual authentication process are as follows:

S1 : Authentication between a mobile-user and home-agent
An MU authenticates HA through a message M_4. Here, the MU device computes $H_B^* = h(H_M \| ID_F \| C_M)$ and verifies $H_B^* \stackrel{?}{=} H_B$ to ensure the legality of FA and HA. Consequently, HA authenticates MU on reception of the authentication message M_2. HA retrieve authentication parameters from blockchain, then computes $M_B^* = h(H_M^* \parallel C_M) \oplus R_M^*$ and verifies $M_B^* \stackrel{?}{=} M_B$ to ensure the legality MU.

S2 : Authentication between home-agent and foreign-agent
HA authenticates FA on receiving M_2. HA calculates $F_B^* = h(F_A \parallel S_{FA})$ and verifies $F_B^* \stackrel{?}{=} F_B$ to check the legality of FA. Similarly, FA authenticates HA through the message M_3. FA retrieve authentication parameters from the blockchain, then computes $H_A^* = h(ID_H \| M_B \| S_{FA})$ and verifies $H_A^* \stackrel{?}{=} H_A$ to ensure the legality of HA and the mobile user, respectively.

Table 4
Summary of mutual authentication and session-key negotiation phase.

Mobile user	Foreign agent	Home agent
Generate R_M $M_A = h(ID_M \| \| N_M) \oplus R_M$ $M_B = h(H_M \| \| C_M) \oplus R_M$ $\boldsymbol{M_1 = \{M_A, ID_H, M_B\}}$		
	Generate R_F $F_A = h(M_A \| \| S_{FA}) \oplus R_F$ $F_B = h(F_A \| \| S_{FA})$ $\boldsymbol{M_2 = \{ID_F, F_A, F_B, M_B\}}$	$S_{FA} = h(ID_F \| \| S_{HA})$ $F_B^* = h(F_A \| \| S_{FA})$; $F_B^* \stackrel{?}{=} F_B$ $H_M^* = h(R_1 \| \| S_{HA})$ $R_M^* = h(H_M^* \| \| C_M) \oplus M_B$ $M_A^* = h(R_1) \oplus R_M^*$ $M_B^* = h(H_M^* \| \| C_M) \oplus R_M^*$; $M_B^* \stackrel{?}{=} M_B$ $R_F = h(M_A^* \| \| S_{FA}) \oplus F_A$ $H_A = h(ID_H \| \| M_B^* \| \| S_{FA})$ $H_B = h(H_M^* \| \| ID_F \| \| C_M)$ $H_C = h(ID_H \| \| R_M^*) \oplus R_F$ Update $C_M = C_M + 1$ $\boldsymbol{M_3 = \{H_A, H_B, H_C\}}$
	$H_A^* = h(ID_H \| \| M_B \| \| S_{FA})$; $H_A^* \stackrel{?}{=} H_A$ $SK = h(M_A \| \| R_F \| \| ID_H)$ $\boldsymbol{M_4 = \{H_B, H_C\}}$	
$H_B^* = h(H_M \| \| ID_F \| \| C_M)$; $H_B^* \stackrel{?}{=} H_B$ $R_F = h(ID_H \| \| R_M) \oplus H_C$ $SK = h(M_A \| \| R_F \| \| ID_H)$; Update $C_M = C_M + 1$		

4.3. Session key fairness and perfect forward secrecy

The proposed mutual authentication framework is designed in such a manner that all communication parties have equal contribution in the session key generation. A session key $SK = h(M_A \| R_F \| ID_H)$, where MU contributes M_A, FA contributes R_F and HA contributes ID_H. Hence, the proposed protocol ensures fairness in the session-key negotiation. In addition, the random numbers R_M and R_F will be freshly produced in each session. Even, if the home agent's private-key S_{HA} is revealed, all foregoing shared session keys remain secure. Therefore, this security framework ensures perfect forward secrecy.

4.4. No verifier tables

In the proposed protocol, the authentication-related information will be stored on blockchain instead of HA and FA verifier tables. The information stored on the blockchain is tamper-proof and decentralized. Therefore, an adversary fails to launch a stolen verifier attack, dictionary attack, an SQL injection attack to retrieve sensitive mobile user information like identities and passwords. Besides, the security system is based on dynamic Diffie–Hellman key exchange to distribute a common secret key in between HA, FA. Due to the hardness of discrete algorithmic problem, it is impossible for the attacker $\mathscr{A}$ to deduce the shared secret key $S_{FA} = (ID_F \parallel S_{HA})$ from a mobility network. Hence, the proposed authentication system prevents stolen-verifier attacks.

4.5. Resist against impersonation attacks

In this scenario, an intruder $\mathscr{A}$ intercepts a sequence of messages M_1, M_2, M_3, M_4 from the communication channel to impersonate MU, FA or HA. However, an attacker encounters various challenges.

4.5.1. MU impersonation attack

An intruder $\mathscr{A}$ should have identity ID_M and password PW_M to cheat the mobile user. In the proposed system, the user credentials have been not transmitted in the authentication sessions. Besides, if an attacker $\mathscr{A}$ guess the MU's password and submits into the authentication system. However, without knowledge of authentication parameters $\{ID_M, N_M, H_M\}$ the adversary cannot make a valid authentication request $M_1 = \{M_A, ID_H, M_B\}$ to cheat FA as well as HA. Therefore, the proposed Mobile-Chain prevents the MU impersonation attack.

4.5.2. FA impersonation attack

With the intractability property of Diffie Hellman key exchange, the intruder is unable to deduce the secret key S_{FA} to cheat HA and MU. Consequently, without knowledge of the random number R_F and S_{FA}, the intruder is unable forge a message $M_2 = \{ID_F, F_A, F_B, M_B\}$. Thus, the proposed system resilience against FA impersonation attacks.

4.5.3. HA impersonation attacks

Without knowledge of HA's private key S_{HA}, the authentication parameters $\{R_1, C_M, H_M\}$ an attacker $\mathscr{A}$ will not be allowed to form a valid authentication response $M_3 = \{H_A, H_B, H_C\}$ to forge MU and FA. Hence, the system prevents the HA impersonation attack.

4.6. Prevention against replay attacks

If an attacker $\mathscr{A}$ intercepts the authentication request messages $\{M_1, M_2\}$ from the public channel to replay the home agent in the next session. However, an adversary could not be replayed to bypass HA's authentication. Because, the mobile device MU, FA generates fresh random numbers $\{R_M, R_F\}$ in each session. Besides, the proposed Mobile-Chain implements a counter based system to prevent replay attacks. If the intruder replays an old message then HA senses the intrusion while retrieving and analyzing the original counter C_M of the mobile user from the blockchain. Therefore, this system withstands replay attacks.

4.7. Prevention against Denial of Service (DoS) attacks

In the existing mobile user authentication systems, an intruder $\mathscr{A}$ inputs wrong credentials during the login phase and forms invalid requests to the server, unfortunately, it could be detected only at HA. $\mathscr{A}$ repeat this process several times to overload the authentication system with invalid requests. Obviously, this process restricts the legal users to gain access to the system and increases the additional overheads on the server, which results in Denial-of-Service attacks. To withstand against DoS attacks, the proposed system is implemented with local password verification. Here, MU device computes $M_P^* = h(ID_M \| PW_M \| N_M)$ and compares $M_P^* \stackrel{?}{=} M_P$ at the client side. If the verification is unsuccessful, the protocol denies access to the system and terminates the authentication protocol. Therefore, the proposed protocol is designed to detect wrong credentials quickly and eliminates additional communication and, computational overheads.

4.8. Clock synchronization problem

The proposed authentication protocol is designed based on the counters C_M instead of timestamps, which require additional clocks at MU, HA and FA to withstand replay attacks. The received timestamp value is compared with the threshold value to check against replay attacks at the receiver side. However, the timestamp-based authentication protocols still suffer from the replay attacks as the delay in the transmission is unpredictable in the global mobility networks due to a node failure (software or hardware issues with the node) or network partitions (communication link failure). In the proposed protocol, the counter value will be incremented as per the mobile user interactions with the home agent (HA). Thus, the proposed Mobile-Chain makes use of counters with the blockchain consensus to tackle the replay attacks and clock synchronization problems.

5. Mobile-chain implementation

In this section, the specific technologies and implementation details are presented to demonstrate the proposed authentication system on the blockchain network. The proposed protocol is implemented using the Ethereum blockchain. One of the most widely used blockchain platforms for developing decentralized applications and smart contract solutions is Ethereum. It supports layer 2 solutions which are crucial because they support scalability and higher throughput without compromising the Ethereum blockchain's integrity, enabling total decentralization, transparency, and security. In addition, the Ethereum is open-source with huge community support and supports for interoperability. Besides, this platform ensures privacy using zero-knowledge proofs.

Initially, the registration of communication parties such as MU, HA, FA, and the authentication functionalities are implemented through the Smart Contracts written in Solidity programming language. Further, the Smart Contracts are compiled using Remix and deployed into the Ethereum blockchain using MetaMask and Ganache.

The proposed system mainly consists of registration and the user authentication phase. During registration, an administrator of the mobile network registers the system on a decentralized blockchain using a system identification number SID. The blockchain checks the existence of SID and creates a Block using smart Contract. After successful registration, blockchain generates a certificate for the mobility network using its private key ($E_{PR}(SID)$). Consequently, the certificate is encrypted using admins public key $C_S = E_{PU}(E_{PR}(SID))$ and transmitted to the network admin. Finally, the mobility network admin decrypts the certificate using the corresponding private-key and distributes the certificate C_S among all other mobility entities such as MU, HA, and FA. The smart contract for mobility system registration with the blockchain is presented in Algorithm 1.

Algorithm 1 Smart contract for the mobility system registration

Result: SID registered with blockchain
Global parameters: *sys*: Object; *BC*: Blockchain
//check for SID on blockchain
if *(SID-exists*$(sys.id, BC) = true$) **then**
  SID already exists on blockchain return error()
else
  register-SID($BC, sys.id$)
end

Upon receiving the certificate C_S from the network admin, mobile devices like MU, FA, and, HA generates a unique certificate called access token using its secret-key. The access token contains the identity of the devices $\{ID_M, ID_H, ID_F\}$, and the associated network information SID provided by the administrator. In addition, the devices of HA, FA, and MU send the access tokens to the blockchain. In this scenario, the Smart contract verifies the legality of the system identifier (SID). If the comparison is successful, the legality of the system is ensured. After, the smart contract allows devices to get registration with blockchain. Eventually, a block of mapping between the system ID (SID) and device ID (EID) will be created and the new certificate called $Auth-pass$ is transferred to the devices for authentication in the future. The smart contract for device registration is shown in Algorithm 2.

Algorithm 2 Smart contract for the device registration

Result: EID registered with blockchain
Parameters: *device*: Object; *sys*: Object; *BC*: Blockchain
//check for SID on blockchain
if *(SID-exists*$(sys.id, BC) = true$) **then**
  The system ID is registered
end
//check for device on Blockchain
if *(EID-exists*$(sys.id, BC) = false$) **then**
  create-mapping($sys.id, device.id, BC$)
else
  return error("does not exist")
end

The devices like HA, FA, and MU can store and retrieve the authentication parameters on the blockchain using $Auth-Pass$ issued in the device registration phase. A smart contract on blockchain triggers a transaction and validates the legality of the devices using $\{SID, EID\}$ present in $Auth-Pass$. If the validation succeeds, the device will be allowed to perform store, retrieve and update operations in the decentralized network. Otherwise, the authentication protocol with blockchain will be aborted. The Smart contract for storing and retrieving the authentication parameters from blockchain using solidity programming as shown in Fig. 2.

The proposed smart contract written in solidity is compiled through remix to get EVM (Ethereum Virtual Machine) byte code. Subsequently, the contract is deployed into the Ethereum blockchain network using the MetaMask. Here, the communication between the remix and MetaMask is achieved through the injected web. Further, a user interface is provided in the proposed protocol to verify the transactions recorded during the mobile user authentication process using a personal blockchain network called ganache.

6. Formal security verification and analysis: simulation study

Formal verification of the system is carried out through the AVISPA tool, which supports the formal and modular language in order to specify the security protocol requirements and properties. In addition, the AVISPA is one of the push-button tools for an Automated Validation of Internet Security Protocols & Application [36]. The objectives of this tool to develop a rich language for specifying threat models and

```
pragma solidity ^0.6.8;
/*Creating a Smart Contract for Mobility-Chain*/
contract Mobile_chain{
struct Home_agent
{
int ID_HA;
string S_HA;
string R1;
int CM;
string HM;
}
Home_agent [] HA;
/*insert MU registration details on Blockchain*/
function insert_Home_agent( int ID_HA, string memory
S_HA, string memory R1, string memory HM, int CM) public
{
Home_agent memory Block=Home_agent(ID_HA, S_HA, R1, CM, HM);
HA.push(Block);
}
/*retrieve authentication parameters*/
function retrieve_Home_agent(int ID_HA) public view
returns(string memory, string memory, string memory)
{
uint i;
for(i=0;i<HA.length;i++)
{
Home_agent memory Block=HA[i];
// Searching for HA's details in Blockchain
if(Block.ID_HA==ID_HA)
{
return(Block.S_HA, Block.R1, Block.HM);
}
}
/*If details not exists in the Blockchain*/
return("Not Found");
}
}
```

Fig. 2. A solidity smart contract for storing and retrieving authentication details from blockchain.

security goals. Besides, AVISPA allow the security organizations to detect the vulnerabilities and threats in the authentication protocols.

AVISPA tool contains four backends to validate the authentication protocols.

- **OFMC**: The "On the fly Model Checker" is designed for bounded verification, and protocol falsification by triggering transitions. The OFMC backend supports the specification of the cryptographic operators, typed and untyped system models.
- **CL-AtSe**: The "Constraint-Logic based Attack-Searcher" performs redundancy elimination and constraint solving methods on cryptographic operations. CL-AtSe handles message concatenation and supports type flaw detection [36].
- **SATMC**: The "SAT-based Model Checker" develops the formula encoding scheme on the security protocol, which represents the violation of the security requirements and functionalities.
- **A4SP**: The "Tree Automata based on Automatic-Approximations for Analysis of Security-Protocol" approximate an adversary capability based on the regular-tree language.

In order to perform security verification, the blockchain-based security framework is modeled in a modular and role-based language called HLPSL (High Level Protocol Specification Language). This formal language supports the specification of structures, intruder models, crypto primitives with their complex properties. Eventually, there is a translator in AVISPA namely, "HLPSL2IF" which automatically translates HLPSL specification into equivalent Intermediate Format (IF). Later, which are in turn fed to one of the backends in AVISPA to display a result. Additionally, the Dolev Yao (DY) attacker model is used in the system to specify the attacker $\mathscr{A}$ capabilities [33].

Initially, the process MU obtains a start signal during transition then the state of MU will be changed from 0 to 1. The variable *State* will be used in HLPSL to maintain the current value of the state. The mobile user makes a request $R1$ to the HA for the registration using a $SND()$ signal. Consequently, the mobile user receives the authentication parameters $\{Hm, C_M, h(\cdot)\}$ from the home agent using the $RCV()$ signal. Besides, the mobile device generates the random nonce R_M to ensure message freshness then compose and sends the authentication request $M_1 = \{ID_H, M_A, M_B\}$ to the foreign agent using the public network. After, FA returns the authentication response $M_4 = \{H_B, H_C\}$ to the mobile user. Finally, MU computes the session key, it is the secret shared between HA and MU. HLPSL role specification of the mobile user is depicted in Fig. 3. In HLPSL language, a role system specifies the basic roles, principals, and the number of sessions bounded in authentication protocols. The commonly used data types in HLPSL are: *const, text,* $symmetric\_key, agent, public\_key, nat$. The basic types are used to provide composition between the communication parties.

The communication entities of the proposed system are modeled in HLPSL namely, $Mobile\_user$, $Home\_Agent$, $Foreign\_Agent$ with the roles of a session, environment, and the security goals. The declaration statement $played\_by\ MU$ represents the role of the mobile user in the HLPSL process. The transition of the form $X = | > Y$ emits the event X and performs an activity Y specified in the composition rules. The property $authentication\_on$ represents the required parameters for the authentication process. Further, the goal property $secrecy\_of\ SK$ specifies that the variable SK remains secure during the communication, explicitly the $intruder\_knowledge$ is specified to analyze the proposed protocol strength.

HLPSL specification covers registration and the authentication scenarios of the security system. The statement $(IDM, p1, \{MU, HA\})$ represents the variable ID_M is only known to an MU and HA using the protocol-id $p1$. The home agent validates the mobile user on $\{H_M, C_M\}$. Simultaneously, MU checks the legitimacy of the HA through a random nonce through M_B. The property $witness(MU, FA, mu\_ha\_rm, RM')$ provides witness to the FA through a random nonce R_M generated freshly by the mobile user in each authentication session.

HLPSL role specification of the home agent and the foreign agent is depicted in Figs. 3 and 4, consequently. Mutual authentication between FA and HA is accomplished using variables $\{ID_F, R_F\}$, and the mutual authentication process between HA and FA is carried out using the variables $\{R_F, H_A\}$. Furthermore, MU, FA and, HA ensure the confidentiality service $\{ID_M, N_M, N_F, SK\}$ through the secret property bounded with different protocol entities.

HLPSL role specification for the session, environment, and security goals as depicted in Fig. 4. In this system, a session is a composition of MU, HA, and FA roles, respectively. The environment in which a security framework is analyzed with an $intruder\_knowledge$ is specified in environment roles. Similarly, the security goals and requirements of the security framework is described in a goal specification.

The proposed system is a composition of four-sessions: players are communication entities and the intruder $\{i, MU, FA, HA\}$ in a first session. Similarly, players $\{MU, i, FA, HA\}$ in the second session. Finally, players $\{MU, FA, i, HA\}$, $\{MU, HA, FA, i\}$ in a session third and fourth, respectively.

The specified HLPSL roles of MU, HA, and FA are executed in the AVISPA tool using ATtack SEarcher (ATSE) backend. The result of the security protocol using AVISPA is presented in Fig. 5. The summary describes the result of the proposed authentication system, which is safe against attacks and satisfies all security requirements in wireless environments. The details include the test vectors and criteria to conclude whether the security framework will be safe or unsafe. Besides, the number of visited-nodes, a depth of attack search with a time as shown in Fig. 5. In addition, the name of the protocol, goals

%MU specification in HLPSL:

role Mobile_user (MU, HA, FA: Agent,
Snd, Rcv: channel(dy))
PK: private key, SK: session key
H: hash_operation; played-by MU
local State: nat,
IDh, IDf, IDm, Nm, Rf', Mp, Cm,
PWm, Hm, R1, Sha: text,
Ma, Mb, Hb, Rm', Hc: text,
const fa_ha_rf', mu_fa_rm',
init State := 0
p1, p2, p3, p4 : protocol id
transition

% MU registration with HA

State: = 0 /\ Rcv(start) =|>

% Send registration request to HA

State': = 1 /\ Nm':= new() /\ R1':= H(IDm.Nm')
/\ Snd({R1'}_PK)/\ secret(IDm, p1, {MU, HA})
/\ secret(Nm', p2, MU)

% Receive registration response from HA

State: = 1 /\ Rcv({H(R1.Sha).Cm.H}_PK)
=|> State': = 2 /\ secret(Sha, p3, HA)

% Login and authentication phase

/\ Rm': = new() /\ Ma': = {H(IDm'.Nm')}.xor Rm'
/\ Mb': = H(Hm'.Cm').xor Rm'

% Send authentication request to FA

/\ Snd(Ma'.Mb'.IDh)
/\ witness (MU, FA, mu_fa_rm, Rm')

% Authentication response from FA

State:= 2 /\ Rcv((H(Hm'.IDf.Cm')).
H(Idh.Rm').xor Rf')=|>

% Compute session-key

State':= 6 /\ SK':= H(Ma'.Rf'..IDh)
/\secret(SK', p3, {MU, FA})

end role

% Home Agent Specification in HLPSL

role Home_agent (MU, HA, FA: agent,
Sha: HA's secret-key, Sfa: FA's secret key
Snd, Rcv: channel(dy)), H: hash_operation,
played_by HA
local State : nat,
R1, Cm, Hm, IDh, IDf, Nm',Rf,
Rm', Ma, Mb, Fa, Fb, Ha, Hb, Hc: text,
const fa_ha_rf, mu_fa_rm , ha_fa_rf,
init State: = 0; p1, p2, p3, p4, p5 : protocol-id

% HA receives registration request from MU.

transition
State = 0 /\ Rcv({H(IDm.Nm')}_SK) =|>
State' := 3 /\ secret(IDm, p1, {HA, MU})
/\ secret(Hm', Cm p2, {MU, HA})

% Send registration response

/\ Hm':= H((IDm.Nm').Sha)
/\ Snd({Hm'.Cm'}_Sha)/\ secret(Sha, p3, HA)

% Receive authentication request M2

State = 3 /\ Rcv(IDf.(H(Ma.Sfa). xor Rf'.
h(Fa.Sfa).H((Hm'.Cm'). xor Rm'))=|>
State' := 5 /\ secret(Sfa, p4, FA)

% Send authentication response M3

/\ Rm' := H(Hm'.Cm').xor Mb'
/\ Rf' =H(Ma'.Sfa). xor Fa
/\ Ha'=H(IDh.Mb'.Sfa)/\Hb'=H(Hm'.IDf.Cm')
/\Hc'=(H(IDh.Rm'). xor Rf')
/\ Snd(Ha'.Hb'.Hc')/\ secret(Sha, p5, HA)
/\ secret(Rm', Cm, {MU, HA})
/\ secret(Rf', Sfa, {FA, HA})
/\ request(MU, HA, mu_ha_idm, IDM)
/\ request(FA, HA, fa_ha_idf, IDF)
/\ witness (MU, HA, mu_ha_rm, RM')
/\ witness (FA, HA, fa_ha_rf, RF')
end role

Fig. 3. HLPSL specification of the mobile user and home agent.

to be achieved, and the back-end details are displayed in the AVISPA Output Format (OF).

The proposed security protocol is animated using a tool called Security Protocol Animator (SPAN) [38]. This tool interactively builds Message Sequence Chart (MSC) for the specified protocol, which can be viewed as MU, FA, HA trace from HLPSL. SPAN consists of a *intruder mode*, which interactively build attacks and displays to the user. Additionally, SPAN maintains the execution-trace corresponding to the protocol execution which allows performing an attack simulation when the attack is encountered in the authentication protocol. A message sequence chart for mobile users and servers with the intruder knowledge of the system presented in Fig. 6 shows that the protocol does not reveal any secret keys.

7. Performance evaluation

The proposed Block-chain based authentication protocol is compared with the well-known and recently proposed authentication systems in the literature to afford roaming services in mobile environ-

```
%Foreign Agent Specification in HLPSL
role Foreign_agent (MU, HA, FA: agent,
Send, Recv: channel(dy))
Sfa: secret key, H: hash_operation,
Played-by FA, local State: nat,
IDh, IDf, Ma, Mb, Hm, Fa, Fb, Ha,
Hb, Hc, SK, Sha, Cm, Nm: text,
p1, p2, p3, p4, p5: protocol-id
const fa_ha_rf, mu_fa_rm,
init State:= 0
%Receive authentication response from MU
transition
State = 0 /\ Rcv((H(IDm'.Nm').xor Rm').
(H(Hm'.Cm').xor Rm').IDh) =|>
State': = 1 /\ secret(Rm', p2, MU)
/\secret(IDm, p1, {MU, HA})
/\ secret(Sha, p3, HA)
% Forwards authentication request M2
/\ Rf' := new()/\ Fa' := H((Ma'.Sfa').xor)
/\ Fb'=h(Fa.Sfa)
/\ Snd(IDf.Fa'.Fb'.H((Hm'.Cm').xor Rm'))
/\ secret(Sfa, p4, FA)
/\ witness (HA, FA, ha_fa_idf, IDf)
/\ witness (FA, HA, fa_ha_rf, Rf')
% Receive Authentication response M3
State = 2 /\ Rcv(H(IDm. H((Hm'.Cm').xor Rm').
Sfa').H(IDf'.Hm'.Cm').H((IDh.Rm').xor Rf'))=|>
State := 3 /\ secret(Sha, p5, HA)
/\ SK':= ((H(IDm'.Nm').xor Rm').Rf'.IDh)
/\ secret(SK', p3, {MU, FA})
% Send Authentication response M4
/\ Snd(H(IDf'.Hm'.Cm').H((IDh.Rm').xor Rf'))
/\ request(FA, HA, fa_ha_rf, Rf')
/\ secret(SK', p3, {MU, FA})
end role
```

```
% Role and goal specification in HLPSL
role session (HA, FA, MU : agent,
SK: session key, H: hash func)
def= local P1, P2, P3, R1, R2, R3 : channel (dy)
composition
Mobile_user (MU, HA, FA, SK, H, P1, R1)
/\ Home_agent (MU, HA, FA, SK, H, P2, R2)
/\ Foreign_agent (MU, HA, FA, H, P3, R3)
end role
role environment ()
def= const ha, fa, mu: agent,
h: hash_operation,
IDh, IDf, Cm, Hm, Rm, Rf : text,
SK: session key,
mu_fa Rm, fa_ha Rf: protocol-id,
p1, p2, p3, p4, p5: proto-id
Intruder_knowledge={mu, ha, fa, h, IDh, IDf, Hm}
composition
session(mu, ha, fa, SK, h)/\ session(i, ha, fa, SK, h)
/\ session(mu, i, fa, SK, h)/\ session(mu, ha, i, SK, h)
end role
goal
secrecy_of p1
secrecy_of p2
secrecy_of p3
secrecy_of p4
authentication_on fa_ha Rf
authentication_on mu_fu Rm
end goal
environment ()
```

Fig. 4. HLPSL specification of the foreign agent, session, environment and goals.

ments. Subsequently, the communication and computational complexity of the proposed security protocol has been evaluated under various measures.

7.1. Security properties comparison

Here, functionalities and security requirements of the proposed security framework are compared with the relevant authentication schemes [7,18,22,37]. The proposed security system is designed to ensure all security requirements in mobility environments with the strength of attack resistance. Table 5 presents the functional and security properties enhanced by the proposed system. It is evident that the protocol is built on decentralized architecture to ensure forward secrecy, user privacy, fairness in session key negotiation, and provides a secure mutual authentication in between MU, HA, FA. In addition, the Block-chain based protocol prevents from various attacks in the wireless and global mobile environment.

7.2. Performance analysis

It is a fact that the devices in mobile and wireless environments are resource-constrained. In fact, these devices will have limited computing capability due to low power, bandwidth, memory, and processor.

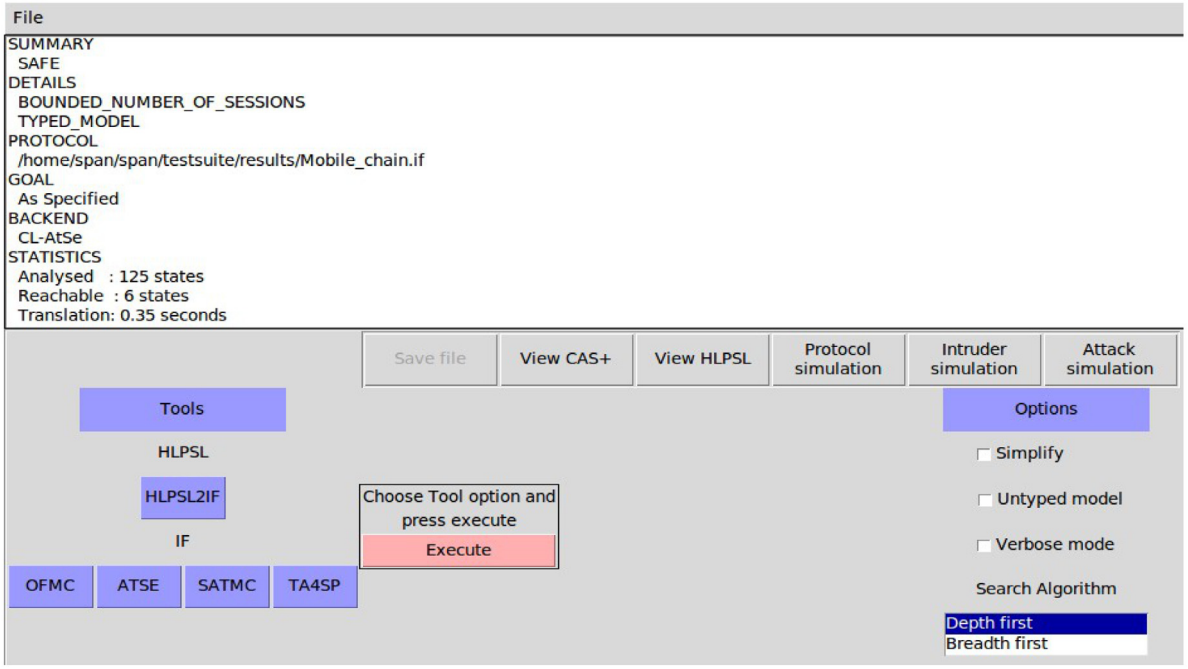

Fig. 5. Result analysis of the proposed system using AVISPA, ATSE back-end.

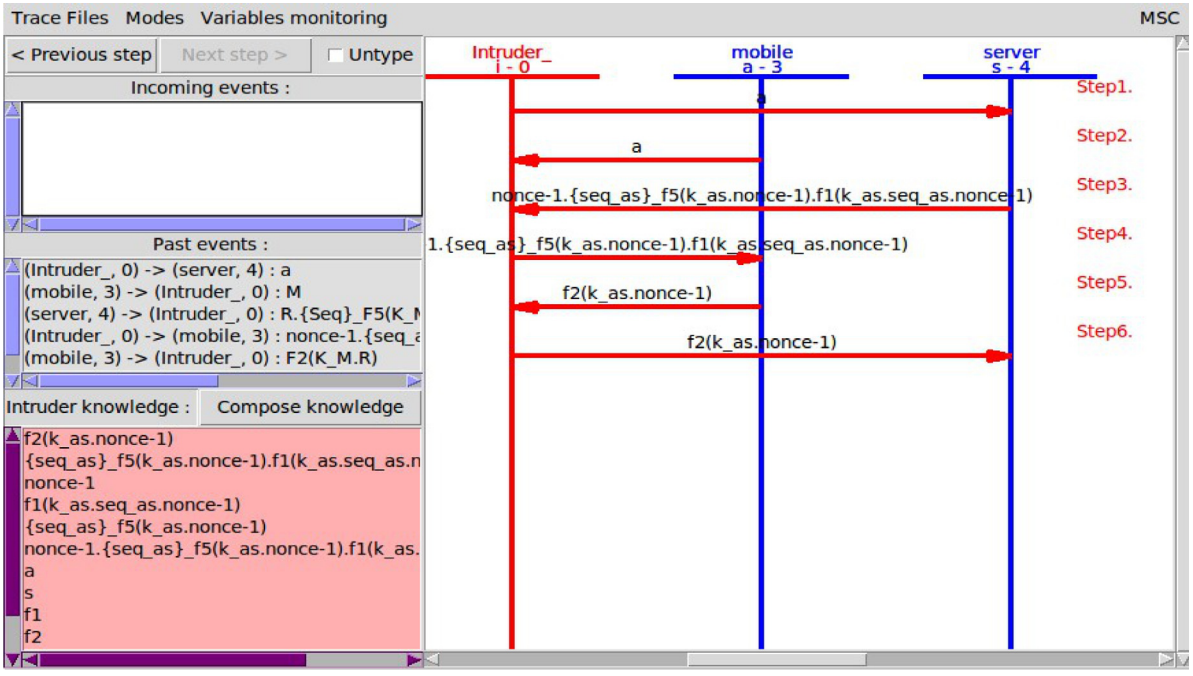

Fig. 6. A message sequence chart between mobile user and server using SPAN.

Hence, it is very crucial to design a lightweight and energy efficient mutual authentication system to preserve the user privacy and security in the mobile and wireless environments.

In general, the performance as well as the efficiency of the security protocol in the mobile network depends on the communication cost, which contains the number of message exchange between MU, HA and, FA. In addition, the computational cost involves the type of cryptographic algorithms used to ensure security services. Therefore, the communication and computational cost will be taken into account while designing the secure authentication system.

The parallel blockchain key derivation function permits the base station transceivers to be ready with the handover key before the handover mechanism is activated by the trigger and the roaming process requires the mobile user to deduce a handover-key and verify for the matching-key with a destination base station. Thus, the roaming process with the blockchain improves the performance of the handover process in the mobility network as compared to the current LTE system [35]. However, the blockchain system performance in the mobility network depends on the probability of adding the block to the chain, and the number of blocks exchanged during the roaming process. The average block time is approximately 10 s in the Ethereum blockchain network [39]. A large number of blocks in the network incurs competition for the communication channel, which results in a lower packet delivery ratio. Eventually, it decreases the probability of a successful block

Table 5
Comparison of the security requirements and functionalities.

Functional & security requirements	Protocol [18]	Protocol [37]	Protocol [7]	Protocol [22]	Proposed
Mutual authentication	✓	✓	✓	✓	✓
Mobile user privacy	✓	×	✓	✓	✓
Prevents insider attack	×	×	✓	✓	✓
Withstand SQL Injection	×	×	×	×	✓
Withstand impersonation attacks	×	×	×	✓	✓
Resilience to bit-flipping attack	×	×	×	×	✓
Withstand stolen-verifier attack	×	×	✓	✓	✓
prevent password-guessing attacks	×	✓	✓	✓	✓
Prevent replay attacks	✓	✓	✓	×	✓
Consensus mechanisms	×	×	×	×	✓
Perfect-forward secrecy	✓	×	×	✓	✓
Anonymity and untraceability	×	×	✓	✓	✓
Fair session-key negotiation	✓	✓	×	✓	✓
Security against DoS attacks	×	✓	✓	✓	✓
Clock-synchronization problem	×	×	×	✓	✓
Decentralization	×	×	×	×	✓
Local password verification	✓	✓	✓	✓	✓

Table 6
Various cryptographic operations along with their execution time.

Notation	Primitive used	Algorithm	Execution-time (S)
T_H	Hash function	Secure Hash Algorithm-256	0.0005
T_{Asym}	Asymmetric system	Elliptic Curve Integrated Encryption	0.0172
T_M	Modular exponentiation	Diffie–Hellman key exchange	0.522
T_{Sym}	Symmetric cryptosystem	Advanced Encryption Algorithm	0.0087
T_P	Point multiplication	Elliptic-curve cryptosystem	0.763

Table 7
Performance evaluation of the proposed and relevant security protocols.

Computation	Protocol [7]	Protocol [18]	Protocol [22]	Protocol [37]	Proposed
MU	$7T_H$	$8T_H + 3T_M$	$6T_H$	$10T_H + 3T_P$	$5T_H$
FA	$4T_H$	$3T_H$	$4T_H$	$5T_H + 2T_P$	$4T_H$
HA	$9T_H + 2T_{Sym}$	$8T_H + T_M + 3T_{Sym}$	$10T_H + T_{Sym}$	$7T_H + 2T_P$	$10T_H$
Total	$\mathbf{20T_H + 2T_{Sym}}$	$\mathbf{19T_H + 4T_M + 3T_{Sym}}$	$\mathbf{20T_H + T_{Sym}}$	$\mathbf{22T_H + 7T_P}$	$\mathbf{19T_H}$
Time (s)	**0.0274**	**2.524**	**0.0187**	**5.353**	**0.0095**

exchange between MU, FA, and HA. Nevertheless, this issue could be addressed using the PB-KDF mechanism in the system-initialization phase [35].

Notably, several crypto algorithms have been simulated on the smartphone to analyze the performance of the Mobile-Chain system in the context of resource-limited mobile environments [40]. The Android operating system with an advanced RISC machines Cortex-A8 processor has been used in the Smartphone. In addition, the frequency of 0.72 GHz is used to perform the simulation. Consequently, several cryptosystems have been implemented through the object oriented programming language using MIRACL (Multiprecision Integer and Rational Arithmetic C++ Library) [41]. This library is mainly designed for securing mobile devices and other embedded devices have smart capabilities. Notably, the hash computation is performed through SHA-256 which is considered as secure than other hash functions. The Advanced Encryption Standard (AES) symmetric cryptosystem with the key-length of 192-bits is used to implement the symmetric cryptosystem. Further, the public-key is cryptosystem has been implemented using ECIES (Elliptic-Curve Integrated Encryption Scheme). Table 6 summarizes the experimental results of various cryptographic algorithms with an execution time.

Table 7 presents the execution time of various authentication protocols. In addition, the number of cryptographic operations required by the user, HA, FA in the mutual authentication phase is summarized. Notably, the mutual authentication and session key negotiation protocol runs frequently than other phases in the authentication process since the protocol provides a single registration for the mobile user.

In the proposed Mobile-Chain, the mobile user acquires five hash operations to form an authentication request M_1 and verifying the authentication response M_4. Subsequently, the foreign agent requires four hash operations to forward the authentication request and authentication response messages $\{M_2, M_4\}$ between HA and the user, respectively. Similarly, the home agent needs ten hash computations to form an authentication response message M_4 in order to provide the mutual authentication and a session key negotiation in between FA and MU, respectively.

It is obvious from Table 7 that the mutual authentication framework needs 0.0095 s to complete the entire authentication and session-key establishment process. Whereas, other protocols in the literature [7,18,22,37] takes more computation time than the proposed protocol. Furthermore, the comparison of the computation overhead with respect to crypto functions used in registration, mutual authentication, and the password change phase is listed in Table 8. Notably, the proposed protocol is designed using a smaller number of light-weight ciphers such as hash operations and the private key cryptosystems. Therefore, this Mobile-Chain framework is light-weight, efficient, and implementable in the resource-limited mobile networks.

The communication cost (bits) of this security protocol and the relevant security protocols [7,18,22,37] for the mobility networks are outlined in Table 9. To analyze the communication cost of the authentication protocols, we used a secure hash algorithm of length 160 bits. Besides, the length of the counter value C_M, timestamps, user information and, the random numbers $\{R_M, R_F, N_M\}$ are 160 bits, respectively. Further, we assumed the length of modular exponentiation and the elliptic curve point multiplication operations are 320 bits, respectively. Notably, the registration request $R_1 = \{h(ID \parallel N_M)\}$ and the response $R_2 = \{H_M, C_M, h(\cdot)\}$ of the proposed security protocol acquires $(160 + 160 + 160 + 160) = 640$ bits, respectively. Consequently, a login message $M_1 = \{M_A, ID_H, M_B\}$ wants $(160 + 160 + 160) = 480$ bits and the authentication request $M_2 = \{ID_F, F_A, F_B, M_B\}$ from FA

Table 8
Comparison of cryptographic operations to analyze computation cost.

Phase	Protocol											
	Protocol [7]		Protocol [18]				Protocol [22]		Protocol [37]		Proposed	
	H	E/D	H	E/D	E	M	H	E/D	H	P	H	E/D
Registration	3	1	5	1	0	0	4	1	5	1	3	1
Login & Auc	20	2	24	3	3	1	20	1	22	7	19	1
Password change	4	0	10	0	0	0	4	0	6	0	2	0
No of operations	**27**	**3**	**39**	**4**	**3**	**1**	**28**	**2**	**33**	**8**	**24**	**2**

H: Hash operation; E/D: Encryption and Decryption operations; E: Exponentiation operation; P: Point multiplication; M: Modular operation.

Table 9
Analysis of communication costs in bits.

Phase	Protocol [7]	Protocol [18]	Protocol [22]	Protocol [37]	Proposed
Registration	640	1120	640	960	640
Login	800	800	480	800	480
Authentication	2560	2400	1440	1760	1440
Total (bits)	**4000**	**4320**	**2560**	**3520**	**2560**

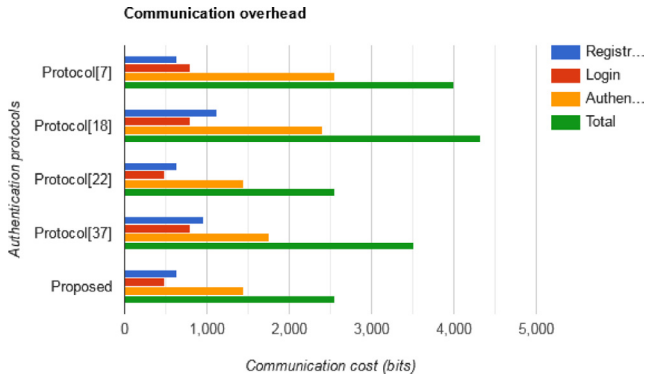

Fig. 7. Comparison of communication overheads.

needs $(160 + 160 + 160 + 160) = 640$ bits. Finally, the authentication response $M_3 = \{H_A, H_B, H_C\}$ from HA and the session-key negotiation message $M_4 = \{H_A, H_B\}$ from FA needs $(480 + 320) = 800$ bits. Thus, the proposed security framework is designed to acquire $(640 + 480 + 640 + 800) = 2560$ bits.

The communication complexity of this security system and the relevant mutual authentication protocols [7,18,22,37] are compared and depicted in Fig. 7. It is obvious that the Mobile-Chain mutual authentication framework possesses low communication overhead. In fact, which requires less number of message exchanges as well as communication bits. Hence, the proposed blockchain-based mutual authentication protocol enhances communication and computational efficiency.

8. Conclusion

In this article, a novel blockchain-based mutual authentication system has been proposed for mobility networks, which is immutable, decentralized, peer-to-peer and distributed in nature. This security system protects user anonymity and resistance to various attacks. Besides, the proposed mutual authentication framework make use of consensus protocol to ensure reliability, safety and fault tolerance in the mobility network. Rigorous security and functional requirement comparison has been done to prove the strength of the authentication system. In addition, formal security verification and validation has been performed through AVISPA using HLPSL. Subsequently, the implementation of the proposed blockchain-based framework has been done on the Ethereum platform using the smart contracts developed in solidity language, which strengthen the security, transparency, and decentralization in the mobile network. Finally, the performance analysis outlines that, the security framework satisfies all functional and security requirements in the context of mobile networks. Further, the protocol is lightweight, efficient, possesses less communication and computational overhead as compare to the recent mutual authentication systems in [7,18,22,37] to provide a roaming service in the mobility environments.

CRediT authorship contribution statement

Indushree M.: Conceptualization, Design, Analysis, Writing – review & editing. **Manish Raj:** Conceptualization, Design, Analysis, Writing – review & editing. **Vipul Kumar Mishra:** Conceptualization, Design, Analysis, Writing – review & editing. **Shashidhara R.:** Conceptualization, Design, Analysis, Writing – review & editing. **Ashok Kumar Das:** Conceptualization, Design, Analysis, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

The authors thank the anonymous reviewers and the editor for their valuable feedback on the paper, which helped us to improve its quality and presentation.

References

[1] C.T. Nguyen, D.N. Nguyen, D.T. Hoang, H.-A. Pham, N.H. Tuong, Y. Xiao, E. Dutkiewicz, BlockRoam: Blockchain-based roaming management system for future mobile networks, IEEE Trans. Mob. Comput. 21 (11) (2022) 3880–3894.
[2] N. Weerasinghe, T. Hewa, M. Dissanayake, M. Ylianttila, M. Liyanage, Blockchain-based roaming and offload service platform for local 5G operators, in: 18th IEEE Annual Consumer Communications & Networking Conference (CCNC'21), Las Vegas, NV, USA, 2021, pp. 1–6.
[3] P. Gope, T. Hwang, Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks, IEEE Syst. J. 10 (4) (2016) 1370–1379.
[4] M. Karuppiah, S. Kumari, X. Li, F. Wu, A.K. Das, M.K. Khan, R. Saravanan, S. Basu, A dynamic ID-based generic framework for anonymous authentication scheme for roaming service in global mobility networks, Wirel. Pers. Commun. 93 (2) (2017) 383–407.
[5] R. Madhusudhan, R. Shashidhara, A secure and lightweight authentication scheme for roaming service in global mobile networks, J. Inf. Secur. Appl. 38 (2018) 96–110.
[6] D. Zhao, H. Peng, L. Li, Y. Yang, A secure and effective anonymous authentication scheme for roaming service in global mobility networks, Wirel. Pers. Commun. 78 (1) (2014) 247–269.
[7] G. Xu, J. Liu, Y. Lu, X. Zeng, Y. Zhang, X. Li, A novel efficient MAKA protocol with desynchronization for anonymous roaming service in global mobility networks, J. Netw. Comput. Appl. 107 (2018) 83–92.
[8] D. He, M. Ma, Y. Zhang, C. Chen, J. Bu, A strong user authentication scheme with smart cards for wireless communications, Comput. Commun. 34 (3) (2011) 367–374.

[9] X. Li, J. Niu, M.K. Khan, J. Liao, An enhanced smart card based remote user password authentication scheme, J. Netw. Comput. Appl. 36 (5) (2013) 1365–1371.
[10] R. Madhusudhan, R. Shashidhara, A secure anonymous authentication protocol for roaming service in resource-constrained mobility environments, Arab. J. Sci. Eng. 45 (4) (2020) 2993–3014.
[11] R. Madhusudhan, Shashidhara, An efficient and secure authentication scheme with user anonymity for roaming service in global mobile networks, in: 6th International Conference on Communication and Network Security (ICCNS '16), Singapore, 2016, pp. 119–126.
[12] R. Madhusudhan, R. Mittal, Dynamic ID-based remote user password authentication schemes using smart cards: A review, J. Netw. Comput. Appl. 35 (4) (2012) 1235–1248.
[13] E. Yoon, K. Yoo, Young, K. Ha, A user friendly authentication scheme with anonymity for wireless communications, Comput. Electr. Eng. 37 (3) (2011) 356–364.
[14] C. Li, C. Lee, A novel user authentication and privacy preserving scheme with smart cards for wireless communications, Math. Comput. Modelling 55 (1) (2012) 35–44.
[15] Q. Jiang, J. Ma, G. Li, Z. Ma, An improved password-based remote user authentication protocol without smart cards, Inf. Technol. Control 42 (2) (2013) 113–123.
[16] F. Wen, W. Susilo, G. Yang, A secure and effective anonymous user authentication scheme for roaming service in global mobility networks, Wirel. Pers. Commun. 73 (3) (2013) 993–1004.
[17] H. Mun, K. Han, Y.S. Lee, C.Y. Yeun, H.H. Choi, Enhanced secure anonymous authentication scheme for roaming service in global mobility networks, Math. Comput. Modelling 55 (1) (2012) 214–222.
[18] M. Karuppiah, R. Saravanan, A secure authentication scheme with user anonymity for roaming service in global mobility networks, Wirel. Pers. Commun. 84 (3) (2015) 2055–2078.
[19] P. Gope, T. Hwang, An efficient mutual authentication and key agreement scheme preserving strong anonymity of the mobile user in global mobility networks, J. Netw. Comput. Appl. 62 (2016) 1–8.
[20] C.-C. Lee, Y. Lai, C. Chen, S.-D. Chen, Advanced secure anonymous authentication scheme for roaming service in global mobility networks, Wirel. Pers. Commun. Int. J. 94 (3) (2017) 1281–1296.
[21] F. Ahmadi, M. Nikooghadam, A secure authentication and session key agreement scheme in global mobile networks preserving user anonymity, J. Electr. Eng. 49 (3) (2020) 965–984.
[22] R. Shashidhara, S. Bojjagani, A.K. Maurya, S. Kumari, H. Xiong, A robust user authentication protocol with privacy-preserving for roaming service in mobility environments, Peer-to-Peer Netw. Appl. 13 (6) (2020) 1943–1966.
[23] A. Al-Qerem, Using raft as consensus algorithm for blockchain application of roaming services for mobile network, Int. J. Artif. Intell. Inform. 3 (1) (2022) 42–52.
[24] B. Mafakheri, A. Heider-Aviet, R. Riggio, L. Goratti, Smart contracts in the 5G roaming architecture: The fusion of blockchain with 5G networks, IEEE Commun. Mag. 59 (3) (2021) 77–83.
[25] C. Esposito, M. Ficco, B.B. Gupta, Blockchain-based authentication and authorization for smart city applications, Inf. Process. Manage. 58 (2) (2021) 102468.
[26] X. Hao, W. Ren, K.-K.R. Choo, N.N. Xiong, A self-trading and authenticated roaming scheme based on blockchain for smart grids, IEEE Trans. Ind. Inform. 18 (6) (2021) 4097–4106.
[27] J.C. Ferreira, C. Ferreira da Silva, J.P. Martins, Roaming service for electric vehicle charging using blockchain-based digital identity, Energies 14 (6) (2021) 1686.
[28] H. Baniata, A. Kertesz, A survey on blockchain-fog integration approaches, IEEE Access 8 (2020) 102657–102668.
[29] H. Baniata, A. Anaqreh, A. Kertesz, PF-BTS: A privacy-aware fog-enhanced blockchain-assisted task scheduling, Inf. Process. Manage. 58 (1) (2021) 102393.
[30] H. Baniata, A. Kertesz, PriFoB: A privacy-aware fog-enhanced blockchain-based system for global accreditation and credential verification, J. Netw. Comput. Appl. 205 (2022) 103440.
[31] R. Madhusudhan, R. Shashidhara, Mobile user authentication protocol with privacy preserving for roaming service in GLOMONET, Peer-to-Peer Netw. Appl. 13 (1) (2020) 82–103.
[32] M.M. Sohail, M. Hassan, K. Mansoor, A. Ghani, K. Jawad, An improved authentication protocol for global mobility network (GLOMONET), in: 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST'20), Islamabad, Pakistan, 2020, pp. 401–406.
[33] D. Dolev, A. Yao, On the security of public key protocols, IEEE Trans. Inform. Theory 29 (2) (1983) 198–208.
[34] A. Paverd, A. Martin, I. Brown, Modelling and Automatically Analysing Privacy Properties for Honest-but-Curious Adversaries, Tech. Report, 2014, https://ajpaverd.org/publications/casper-privacy-report.pdf. Accessed on September 2022.
[35] H. Lee, M. Ma, Blockchain-based mobility management for 5G, Future Gener. Comput. Syst. 110 (2020) 638–646.
[36] D. Basin, S. Mödersheim, L. Vigano, OFMC: A symbolic model checker for security protocols, Int. J. Inf. Secur. 4 (3) (2005) 181–208.
[37] A.G. Reddy, A.K. Das, E.-J. Yoon, K.-Y. Yoo, A secure anonymous authentication protocol for mobile services on elliptic curve cryptography, IEEE Access 4 (2016) 4394–4407.
[38] Y. Glouche, T. Genet, O. Heen, O. Courtay, A security protocol animator tool for AVISPA, in: ARTIST2 Workshop on Specification and Verification of Secure Embedded Systems, Pisa, Italy, 2006, pp. 1–7.
[39] S. Kim, Impacts of mobility on performance of blockchain in VANET, IEEE Access 7 (2019) 68646–68655.
[40] W. Dai, Crypto++ library 5.1-a free C++ class library of cryptographic schemes, 2011, http://www.cryptopp.com/. Accessed on March 2022.
[41] S. Muftic, E. Hatunic, CISS: Generalized security libraries, Comput. Secur. 11 (7) (1992) 653–659.

Indushree M is Research Scholar in Department of Computer Science Engineering & Technology, Bennett University, Greater Noida, India. She completed her B.Tech in Information Science and Engineering and M.Tech in Information Technology from Visvesvaraya Technological University, India. Her area of interest includes remote user authentication, blockchain, information and network security, wireless and mobile networks, and cross-site scripting.

Manish Raj obtained his Ph.D. Degree from the Indian Institute of Information Technology, Allahabad, India, in Robotics & AI department. Currently, he is Associate Professor at Bennett University, Greater Noida, India. His research interests are Internet of Things, control systems, nonlinear dynamics, humanoid robotics, Blockchain, artificial intelligence, soft computing, and hybrid systems. He published his research work in various SCI/SCIE and Scopus indexed journals and also presented his work in different National & International Conferences which show his research quest. He also served as reviewer of several prestigious and peer reviewed journals.

Vipul Kumar Mishra received his Ph.D. Degree from the Indian Institute of Technology (IIT), Indore, India, in Computer Science and Engineering. Currently, he is Associate Professor at Bennett University, Greater Noida, India. His research interests are Internet of Things, optimization in EDA, CAD for VLSI and Nanotechnologies, emerging technologies swarm optimization, and nature inspired optimization algorithm. He has published his research work in various SCI/SCIE and scopus indexed journals, and also presented his work in different national and international conferences which show his research quest. He also served as reviewer of several prestigious and peer reviewed journals.

Shashidhara R received his PhD in Cryptography and Network Security from National Institute of Technology, Karnataka, Surathkal, India, and the M.Tech degree in communication and networks from the Visvesvaraya Technological University, Belgaum, Karnataka, India. Currently, he is working with the Blockchain R&D, Wipro Technologies, Bengaluru, Karnataka, India. Previously, he was working as an Assistant Professor in the School of Engineering & Applied Sciences, Bennett University (Times of India Group), Greater Noida, India. He has various research publications in conferences indexed in CORE ranking and many journals of international repute, including ACM, Elsevier, Springer, IEEE. He is a reviewer for the many reputed international journals. His research interests include design of robust authentication protocols for wireless and mobility environments, Blockchain technology, cross-site scripting attacks and security in Internet of Things.

Ashok Kumar Das received a Ph.D. degree in computer science and engineering, an M.Tech. degree in computer science and data processing, and an M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently an Associate Professor with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad, India. He was also working as a visiting faculty with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA 23435, USA. His current research interests include cryptography, network security, security in vehicular ad hoc networks, smart grids, smart homes, Internet of Things (IoT), Internet of Drones, Internet of Vehicles, Cyber–Physical Systems (CPS) and cloud computing, intrusion detection, blockchain and AI/ML security. He has authored over 320 papers in international journals and conferences in the above areas, including over 275 reputed journal papers. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He has been listed in the Web of Science (Clarivate^{TM}) Highly Cited Researcher 2022 in recognition of his exceptional research performance. He was/is on the editorial board of IEEE Systems Journal, Journal of Network and Computer Applications (Elsevier), Computer Communications (Elsevier), Journal of Cloud Computing (Springer), Cyber Security and Applications (Elsevier), IET Communications, KSII Transactions on Internet and Information Systems, and International Journal of Internet Technology and Secured Transactions (Inderscience), and has served as a Program Committee Member in many international conferences. He also severed as one of the Technical Program Committee Chairs of the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, June 2019, International Conference on Applied Soft Computing and Communication Networks (ACN'20), October 2020, Chennai, India, and second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, October 2020. His Google Scholar h-index is 73 and i10-index is 207 with over 14,980 citations. He is a senior member of the IEEE.

Vivekananda Bhat K received the Ph.D. degree in computer science and engineering from IIT Kharagpur, India, and the M.Tech. degree in systems analysis and computer applications from the National Institute of Technology Karnataka, Surathkal, India. He is currently an Associate Professor with the Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, India. He has published several articles in reputed Web of Science indexed international journals and conferences. His research interests include cryptology and cyber security. He is a senior member of the IEEE.