Frontiers in Health Informatics
ISSN-Online: 2676-7104
www.healthinformaticsjournal.com
2024; Vol 13: Issue 3
Open Access

HMAM: Hash-Chain Based Multi-Channel Authentication Mechanisms for Edge/Fog Devices

[1]Asha H P, [2]Diana Jeba Jingle
[1,2]Department of Computer Science and Engineering, Christ University, Bangalore, India.
asha.hp@res.christuniversity.in, diana.jebajingle@christuniversity.in

Cite this paper: Asha H P, Diana Jeba Jingle (2024) HMAM: Hash-Chain Based Multi-Channel Authentication Mechanisms for Edge/Fog Devices *Frontiers in Health Informatics, 13* (3), 6688-6700

Abstract: *With the emerging trends in home automation, office automation and industry automation using various IoT devices, there are more and more security vulnerabilities identified and misused. Some of these security compromises in applications like smart cars or military applications can be fatal and prove too costly but these IoT/Fog devices are absolutely needed. Though fog devices have limited computation power, it has the ability to communicate in multiple channels like Cellular Network, Bluetooth, etc. Traditional approaches like password-based authentication, identity based key encryption etc. were found less competitive in terms of computing power, lengthy computation, and too many messages exchanged which can lead to latency problems. In this paper, we propose a novel Hash-chain based Multi-channel Authentication Mechanism (HMAM) which utilizes multiple channels for authentication of fog/IoT devices. The client's metadata is verified using a challenge-response key exchange hash-chain model and then stored on a centralized cum distributed secured blockchain. The proposed method requires less computational time for the resource and computation power constrained devices, as we use multiple channels for mutual authentication for easy and fast communication to avoid data theft or modification during delayed communication thereby reducing latency. Moreover, the proposed HMAM is implemented with very few computations and very few key exchanges between Fog nodes and the server so that security can be achieved in fewer computations. The metadata and node Id are stored in blockchain, and the keys are not stored anywhere in the memory and as a result the storage overhead is reduced.*

Key words: *Fog computing, Multi-channel, Blockchain, Hash chain, Game theory.*

1. Introduction

Over two decades, cloud computing became very popular and many organizations moved from on-premises to cloud for reducing the overhead of managing data centers, cooling, power, manpower, capital investments etc. On the other hand, the popularity of IoT devices over the years such as smart watches, smart home appliances, self-driving cars and even medical and industrial appliances etc. had to be connected to the cloud for many reasons such as automated decision making, status tracking etc. Cloud servers were thousands of miles away from edge devices and many times across countries and continents. This led to latency and many real time decisions could not be made which is trivial for operating of the IoT devices/Fog Nodes such as smart cars and many mission critical medical and industrial applications [29], [30]. To mitigate the latency challenges associated with traditional cloud computing for Internet of Things (IoT) devices, Cisco pioneered the concept of fog computing. This intermediate layer positions servers closer to IoT devices at the network's edge, reducing the physical distance and latency [3]. The terms "edge computing" and "fog computing" are often used interchangeably in the literature. This decentralized computing architecture brings processing, storage, and intelligence closer to IoT devices. It effectively extends the capabilities of cloud computing, including design,

network, and storage, to the edge of the network. By reducing the distance between IoT devices and the cloud, fog computing enhances efficiency in data analysis, processing, and storage [1]. Key characteristics of fog computing include low latency, location awareness, mobility, wide geographic distribution, high scalability, wireless access, real-time applications, and heterogeneity [9]. Advantages of fog computing include improved security, privacy, reduced bandwidth and operational costs, low latency, business agility, mobility, remote deployment, and better data handling [10], [11]. Fog servers act as intermediaries between fog nodes (IoT devices) and the cloud, facilitating real-time decision-making. Edge servers, unlike cloud servers, operate in challenging environments, often exposed to extreme temperatures (-30°C to +50°C). This limits their physical security and computing power, making them susceptible to theft and tampering. To compensate for these constraints, edge servers are designed to handle time-sensitive tasks locally while offloading computationally demanding workloads to the cloud. These devices typically rely on wireless networks (like 3G/4G) for connectivity, further impacting their performance compared to cloud-based solutions [2].

The rapid evolution of cellular networks, particularly with the advent of 5G technology, has significantly accelerated data transfer speeds. While theoretical 5G speeds can reach up to 10Gbps, real-world average speeds often exceed 150mbps. This surge in bandwidth has led to reduced latency, enabling faster and more efficient cloud connectivity [6]. IoT devices leverage various communication channels, such as Bluetooth, ZigBee, Cellular, and Wi-Fi, to connect with Edge Devices [5]. The increasing network speeds enabled by 4G and 5G have fueled the development of advanced IoT devices [31], [32], including smart cars and cashless payment systems. However, a major challenge for Edge Servers and IoT devices remains security [7].

Sybil attacks, where malicious entities impersonate legitimate nodes, pose a significant threat to fog computing systems [8]. These attacks can disrupt network operations and compromise data integrity. To mitigate such risks, secure mutual authentication between devices is essential. Unlike cloud servers, edge servers and IoT devices often lack the computational resources and security infrastructure necessary for traditional attack detection and prevention. This necessitates the development of specialized security solutions tailored to the unique constraints of fog environments [4].

This paper proposes a novel authentication scheme that leverages blockchain technology to enhance the security of fog computing systems. The proposed scheme, known as HMAM, employs two-channel authentication and hash-chain mechanisms to establish secure communication between nodes and servers. A key aspect of HMAM is its focus on securing metadata [6], which is essential for identifying and managing data within fog networks. By storing metadata on a distributed blockchain, the proposed scheme ensures its integrity and prevents unauthorized access [15]. Blockchain's decentralized nature [12], cryptographic hash proofs [18], [19], and consensus-based approach [16] provide strong security guarantees and facilitate efficient data validation and tracking [17]. In conclusion, HMAM offers a promising solution to the security challenges faced by fog computing systems [13], [20]. By combining two-channel authentication, hash chain-based protocols, and blockchain technology, HMAM provides a robust and efficient mechanism for protecting against sybil attacks and ensuring the integrity of data in fog networks [14].

1.1 Motivation

Despite the widespread adoption of cloud computing in research, its infrastructure remains substantial and vulnerable to security threats. The proliferation of devices like mini computers and laptops in homes and workplaces generates large volumes of sensitive data that demand immediate and secure attention. The metadata associated with this data, which is crucial for its efficient use and retrieval, is also a critical concern within the network. Traditional security measures, such as password-based authentication and identity-based key

encryption, have proven inadequate to address these challenges. This paper proposes a novel matrix-based key exchange model for authentication of fog nodes. This approach aims to enhance the security of fog nodes by mitigating various security attacks, reducing latency, and minimizing computational and storage overhead.

1.2 Contribution

We propose a secure node authentication protocol based on game based key exchange protocol. We introduce a novel authentication protocol for fog edge devices that leverages a game-based key exchange approach. Our scheme offers several advantages:

- Efficient Mutual Authentication: Through a matrix-based key exchange model, we ensure mutual authentication while minimizing computational overhead, making it suitable for resource-constrained edge nodes.
- Enhanced Performance: By utilizing two-channel addresses, we address the latency issues associated with existing single-channel schemes. This approach also reduces computational burden and storage requirements.
- Robust Security: Keys are not stored locally, eliminating the risk of compromise and the need for secure storage.
- Tamper-Proof Metadata: To verify the authenticity of Fog Clients (FCs), we store their metadata on a tamper-proof blockchain database. During authentication, payoff matrices are exchanged via two channels registered in this metadata, ensuring integrity and security.

In summary, our proposed protocol offers a secure, efficient, and scalable solution for authenticating fog edge devices, addressing the challenges posed by traditional methods.

1.3 Organization of the Paper

This paper is structured as follows: Section 2 provides a comprehensive review of existing hash chain and blockchain-based system models, along with a detailed analysis of their security requirements. Section 3 outlines our proposed methodology and introduces the fundamental building blocks of our system. Section 4 delves into the implementation details, presents the results of our performance evaluation, and offers a thorough analysis of the findings. Section 5 concludes the paper by summarizing our key contributions and discussing future research directions.

2. Literature Review

Numerous studies have explored security mechanisms for edge and fog nodes, employing techniques like blockchain, cryptography, game theory, and key exchange. To delve into these approaches, we conducted a comprehensive survey of existing literature on Fog/Edge security and mutual authentication using blockchain and cryptographic hash chain key exchange models. This section outlines the identified shortcomings in current methodologies and demonstrates how our proposed solution can effectively address these limitations. Authentication in real-time processing systems, especially those with limited computing resources, presents unique challenges. Traditional authentication methods often prove inadequate due to their computational demands.

Imine et al. [21] leverages a centralized cloud-based password verification technique for authenticating fog nodes, but it remains susceptible to brute force and dictionary attacks. Additionally, the reliance on a central password repository poses risks of data breaches and compromised accounts, particularly if users employ weak or reused passwords. We address these shortcomings, by introducing a decentralized authentication mechanism. Our approach eliminates the need for central key storage, ensuring enhanced security. By generating and discarding keys on-the-fly, we mitigate the risks associated with compromised storage. This decentralized

architecture not only strengthens security but also aligns with the resource-constrained nature of real-time processing systems.

Pardeshi et al. [22] introduced a mutual authentication scheme for distributed fog networks that leverages hash chains and zero-knowledge proofs. While their approach offers security, it demands significant computational resources and introduces latency due to multiple message exchanges. To address these limitations, we propose a more efficient scheme that employs a mixed-strategy key exchange hash-chain model. Our approach enables mutual authentication between fog edge devices with minimal computational overhead, making it suitable for resource-constrained environments.

Jia et al. [23] highlight a critical challenge: 5G's high-bandwidth and low-latency channels can lead to network congestion when authentication requests overwhelm the processing capacity of central authentication centers. They address this issue by proposing a decentralized approach using blockchain and sidechain technology. This architecture distributes edge nodes to decentralize the authentication process. Instead of relying on traditional public key infrastructure (PKI), the proposed scheme employs identity-based cryptography (IBC) to reduce processing overhead. However, PKI's reliance on a central server and certificate authority introduces risks, such as key disclosure and the need for secure communication channels. The proposed method mitigates these concerns by using two-channel communication, improving latency and reducing the risk of key compromise. Additionally, to safeguard against malicious attacks when new devices join the network, a client-server handshake mechanism is implemented using game theory and hash chain-based key exchange.

Xuejiao Liu et al. [25] propose a multi-authority attribute-based signature (MABS) system that effectively manages vehicle keys. Their MABS system uses fine-grained policies to ensure secure access control. However, traditional attribute-based encryption (ABE) schemes can be computationally expensive. To address this, our approach uses a lightweight AES encryption only once at the end of a game challenge. This significantly reduces computational overhead compared to ABE. Additionally, we employ monotonous attributes to control user access, allowing data owners to encrypt data using authorized users' public keys.

Cryptographic techniques like proxy re-encryption [26] offer an effective security solution for fog-of-things communication. The work in [26] presents a secure mutual authentication scheme for edge-fog-cloud networks using hash chains and symmetric encryption/decryption, efficiently applicable to smart cards and devices. However, the short length of the master secret key in distributed applications poses a significant vulnerability for long-lived keys. To mitigate these security risks, our proposed method employs a lengthy crypto-string derived from hashed exponential n as the key. This enhanced key length significantly reduces the susceptibility to various security attacks.

We identified the following shortcomings in current technologies: i) Centralized authentication: Reliance on central password repositories exposes systems to risks like brute force attacks, data breaches, and compromised accounts. ii) Computational overhead: Traditional authentication methods often require significant computational resources, making them unsuitable for resource-constrained edge and fog nodes. iii) Increased latency: High-bandwidth, low-latency 5G networks can become overwhelmed with authentication requests, leading to delays and performance degradation. iv) Key management challenges: Managing and protecting cryptographic keys in distributed systems can be complex and error-prone.

Our proposed approach addresses these limitations in the following manner: i) Decentralized authentication: Eliminating the need for central key storage enhances security and mitigates risks associated with compromised repositories. ii) Efficient key exchange: Employing a mixed-strategy key exchange hash-chain model reduces computational overhead and makes authentication suitable for resource-constrained environments. iii)

Distributed authentication: Decentralizing the authentication process using blockchain and sidechain technology distributes the load and reduces the risk of network congestion and reduced latency. iv) Enhanced key management: Using crypto-strings derived from hashed exponential n as keys provides longer, more secure keys, reducing the risk of attacks. By addressing these limitations, the proposed approach offers a more secure, efficient, and scalable solution for authentication in edge and fog computing environments.

3. Methodology

In this section, we delve into the operational mechanics of our proposed HMAM system. This system consists of two distinct phases: the Bootstrapping Phase and the Authentication Phase. A visual representation of the HMAM's architectural framework is provided in Figure 1.

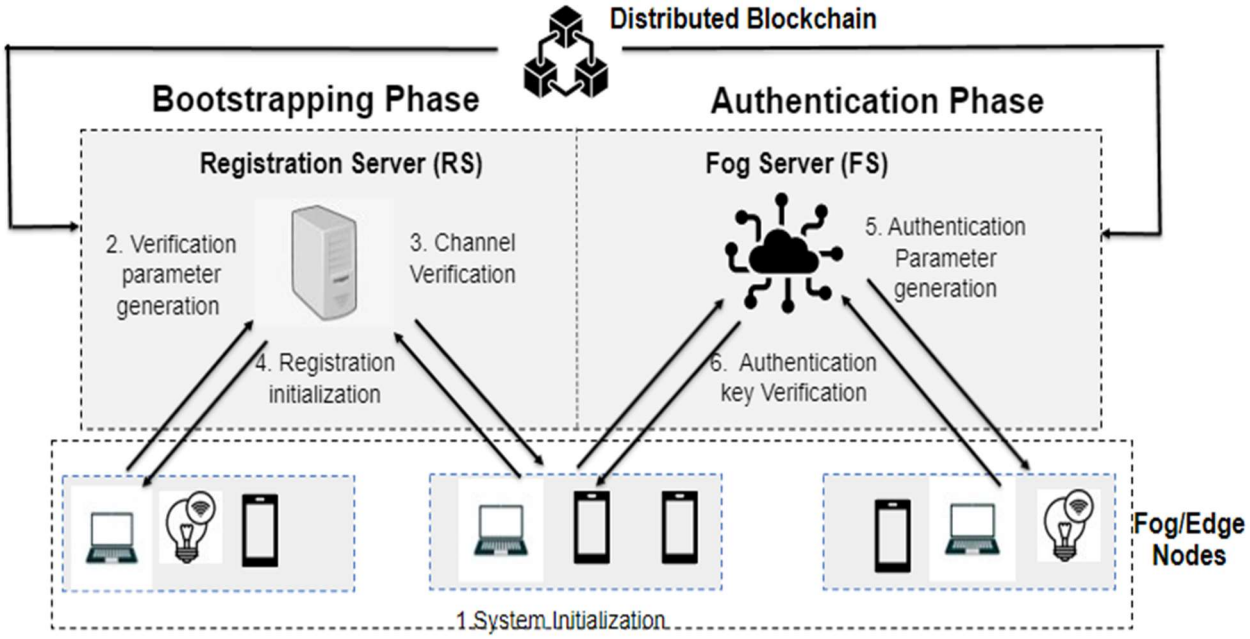

Figure 1. Architectural diagram of the Proposed Method

3.1 Bootstrapping Phase

This is the initial step of the protocol that initiates a process by initializing the client's metadata. When a client or user 'FC' desires to establish a secure connection with a Fog Server (FS), they begin by providing their metadata to the Registration Server (RS). The fog client FC requests registration to the RS by transmitting its metadata via a Wi-Fi channel (channel 1). This interaction can be represented as:

$FC\ \ Wi-Fi\ channel \rightarrow\ \ RS$: Metadata

Upon receiving FC's metadata, the RS verifies the client's Bluetooth channel (Channel 2) by sending a token. This channel verification step is essential for security.

$RS\ \ bluetooth\ channel \rightarrow\ FC: T_{FC\_i}$

FC responds by sending the received token back to the RS over the Wi-Fi network, completing the verification process.

$FC\ \ Wi-Fi\ channel \rightarrow\ RS: T_{FC\_i}$

If the RS successfully receives the token, it authorizes and validates the user. RS generates a unique node ID

for the client and transmits it to FC via Bluetooth channel, along with the assigned Fog Server ID (FS_ID_i).

$$RS \ \ bluetooth\ channel \rightarrow \ \ FC: FC\_ID_i || FS\_ID_i$$

After successful registration, FC's metadata and allocated Node IDs are stored on a distributed blockchain. A pool of fog servers and their corresponding fog nodes are also stored on the blockchain. Each FC stores 128 bits of metadata for each node. This metadata includes 32-bit IP address (IP_i), 48-bit MAC address (MAC_i) and 48-bit Bluetooth address (BT_i) which can be expressed as $Metadata: Hex\big((MAC_i + BT_i + IP_i) \oplus TS_i\big)$. When FC later approaches their assigned FS for authentication, the FS validates FC's information by consulting the distributed blockchain. This ensures the security and integrity of the authentication process.

3.2 Authentication using matrix-based key exchange model

Every fog client/user (FC) seeking a secure connection with a Fog Server (FS) must undergo the Authentication Phase. This phase employs a matrix-based key exchange model with a hash chain to authenticate the fog nodes. The step-wise procedure of authentication phase is described below:

Each FC before establishing a secure connection with FS, approaches the FS for authentication, presenting their identity via Wi-Fi channel.

$$FC \ \ Wi-Fi\ channel \rightarrow \ \ FS: Authentication\ Request$$

The FS retrieves FC's metadata from the distributed blockchain and validates its identity through Bluetooth channel. Additionally, the FS generates a 4x4 matrix $[A]$ and shares it with FC via the same channel.

$$FS \ \ bluetooth\ channel \rightarrow \ \ FC: Authentication\ Response, [A]$$

The FC in return, generates a 4x4 matrix $[B]$ and shares it with the FS via Wi-Fi channel.

$$FC \ \ Wi-Fi\ channel \rightarrow \ \ FS: [B]$$

Both FS and FC performs a XOR operation on their respective matrices to obtain payoff matrices $[payoff_{FS}]$ and $[payoff_{FC}]$. This is expressed as, $[payoff_{FS}] = bitwise\_XOR([A],[B])$ and $[payoff_{FC}] = bitwise\_XOR([A],[B])$. These payoff matrices are then reduced to obtain single values (V_{FS} and V_{FC}) using a game-based mixed dominance strategy. The FS also generates a private Authentication Key (K_{FS}) using the operation:

$$K_{FS} = (MAC_i + BT_i) \oplus TS_i$$

The value V_{FS} and K_{FS} are hashed n times (where $n == V_{FS} == V_{FC}$) and encrypted before being sent to FC via Bluetooth. This is expressed as:

$$FS\ bluetooth\ channel \rightarrow \ \ FC: ENC\big(HASH^n(V_{FS} \,||\, (K_{FS})\big)$$

The FC decrypts the message using its private key (K_{FC}) and sends back a message encrypted with its public key (K_{FS}) and the time elapsed ($T_{elapsed}$) to FS. This time it chooses the Wi-Fi channel.

$$FC \ \ Wi-Fi\ channel \rightarrow \ \ FS: ENC\left(HASH^n(V_{FC} \,||\, K_{FC} || T_{elapsed})\right)$$

Now the FS verifies the message sent by FC and establishes a secure connection between FC and the FS on successful authentication.

3.2.1 Game Based Mixed Dominance Method

The Payoff matrix $[payoff_{FS}]$ is reduced using a game based mixed dominance method, which is explained in the algorithm given in TABLE 1.

TABLE I. Game Based Mixed Dominance Method

Algorithm: Game Based Mixed Dominance Method
Input: $[payoff_{FS}] = bitwise\ XOR([A],[B])$ where $[A]$ and $[B]$ are two matrices $minr$: $the\ minimum\ elements\ of\ each\ row\ of\ the\ payoff\ matrix$

$minr\_max$: $the\ maximum\ of\ row\ minimum$
$maxc$: $the\ maximum\ elements\ of\ each\ column$
$minc\_max$: $the\ minimum\ of\ column\ maximums$

Output: V_{FS}

\# $Step\ 1$: $Initialize\ row\ and\ column\ sums\ of\ the\ payoff\ matrix\ [payoff_{FS}]$
$row\_sums\ =\ sum\ of\ elements\ in\ each\ row\ of\ [payoff_{FS}]$
$col\_sums\ =\ sum\ of\ elements\ in\ each\ column\ of\ [payoff_{FS}]$

$\#step\ 2$: \# $Find\ minimum\ row\ and\ maximum\ column$
$min\_row\_index\ =\ index\ of\ the\ row\ with\ the\ minimum\ sum$
$max\_col\_index\ =\ index\ of\ the\ column\ with\ the\ maximum\ sum$

$\#Step\ 3$: $Remove\ minimum\ row\ and\ maximum\ column$
$[payoff_{FS}] =\ remove\ row\ at\ min\_row\_index\ from\ [payoff_{FS}]$
$[payoff_{FS}] =\ remove\ column\ at\ max\_col\_index\ from\ [payoff_{FS}]$

\# $Step\ 4$: $Check\ for\ Saddle\ Point$
$If\ min\_row\_index\ ==\ max\_col\_index$
"$The\ game\ has\ a\ Saddle\ point$"
$n\ =\ min\_row\_index$
$V_{FS} = n$
$Else$
"$The\ game\ has\ no\ Saddle\ point, go\ to\ Step\ 5$"
\# $Step\ 5$: $Repeat\ steps\ 2-5\ until\ the\ matrix\ is\ reduced\ to\ a\ single\ value\ V_{FS}$
$if\ [payoff_{FS}]\ is\ a\ 1x1\ matrix$:
$V_{FS}\ =\ [payoff_{FS}]\ [0][0]$

This same procedure is adapted by FC and the authentication keys K_{FC} and K_{FC} are saved in a distributed database and hashed n number of times and exchanged between both FC and FS. Thus the proposed key exchange protocol involves a two-channel approach, where communication takes place over two separate channels. This design enhances security by making it more difficult for an attacker to intercept or manipulate the exchanged messages. TABLE 2 describes the notations with their meanings used in the proposed method.

TABLE II. Parameters used for authentication

Notations	Meaning
RS	Registration Server
FS	Fog Server
FC	Fog Client
MAC_i, BT_i, IP_i	MAC address, Bluetooth address, IP address of ith fog node
TS_i	Time stamp used for key generation
T_{FC_i}	Token of ith fog node
FC_ID_i, FS_ID_i	Fog Client ID, Fog Server ID

[A], [B], $[payoff_{FS}]$, $[payoff_{FC}]$	Matrix A, Matrix B, Payoff Matrix of fog server, Payoff Matrix of fog client
V_{FS}, V_{FC}	Value obtained from dominance property
$HASH^n$	n times Hash chain with $n == V_{FS} == V_{FC}$
K_{FS}	FS's private key
K_{FC}	FC's private key

4. Experimental Setup and Analysis

We deploy the workstation using Intel core i5 – 4 gen @ 1.6GHz with 12 GB RAM on a Windows 10 operating system as a Server communicating and two clients with Intel core i5 – 11 gen @ 3 GHz with 8GB RAM on Windows 10 operating system. The server perform two-channel communication with the clients. We use python libraries, functions and modules like asyncio, random, websockets, hashlib, math etc. to implement the proposed model.

4.1 Computation Cost

The proposed matrix-based key exchange model employs a matrix-based approach in which matrices are generated and exchanged among the FC and FS through two different channels. In this way the FS verifies the identity of the FC to achieve efficient mutual authentication, making it well-suited for resource-constrained edge nodes. This method minimizes computational overhead while ensuring the integrity and security of the authentication process.

Each FC sends a matrix [A] via a primary channel (Wi-Fi) to the FS and in return, the FS will send another matrix [B] via a secondary channel (Bluetooth MAC) as registered in the metadata database during the registration phase (already mentioned in authentication phase). As the communication is not on the same channel, the FS must ensure that it is talking to the correct fog node as recorded in the Metadata. To do so, both FS and FC require a common key and a common nonce value. Hence we use the matrix-based key exchange model to compute the common keys (K_{FS} and K_{FC}) and the common nonce values (V_{FS} and V_{FC}) for both FS and FC. The FS computes authentication key K_{FS} and a nonce value V_{FS} and sends it to FC by encrypting it as: $ENC\big(HASH^n(V_{FS} \,||\, K_{FS})\big)$. The FC decrypts the message and computes K_{FC} and a nonce value V_{FC} and sends it to FS by encrypting it as: $ENC\left(HASH^n\big(V_{FC} \,||\, K_{FC} || T_{elapsed}\big)\right)$. The FS decrypts the message on successful authentication.

If a malicious user acquires the authentication key, they can potentially intercept and decrypt messages exchanged between the FC and the FS. However, they cannot successfully authenticate themselves unless they also have control over the secondary channel. Even if a malicious user can decrypt the encrypted message sent by the FC, they cannot provide the corresponding message on the secondary channel. This mismatch will trigger a rejection of the authentication attempt. The secondary channel message must be received within a specific timeframe. If there is a significant delay or the message is not received at all, the authentication will fail.

In bootstrapping phase, the FC makes one request to the RS to join the network by providing its metadata. It receives a token from RS through primary channel (blue tooth) and sends back the token through a secondary channel (Wi-Fi). The authentication process involves several key operations, each with associated computational costs. To generate a 4x4 matrix, 16 elements are involved. This is a relatively simple operation, and the computational cost is negligible. To perform a XOR operation on two 4x4 matrices involves 16 element-

wise XOR operations. This operation is also computationally inexpensive. To reduce the payoff matrices to a single value using a game-based mixed dominance strategy can be more computationally intensive, especially for larger matrices. However, for 4x4 matrices, the computational cost is still manageable. Moreover, during this phase two computations are carried out verify a FC: $ENC\left(HASH^n(V_{FS} \,||\, K_{FS})\right)$ and $ENC\left(HASH^n(V_{FC} \,||\, K_{FC}||T_{elapsed})\right)$.

Hashing the values n times, can be computationally expensive, especially for large values of n. The choice of hash function and its implementation can significantly impact the computational cost. Therefore our model takes O(n* h(m)) computations, where h(m) is the computational cost of the hash function for a message of size m. Encrypting and decrypting messages using public-key cryptography can be computationally expensive, especially for large messages and strong cryptographic algorithms. The choice of encryption algorithm and key length will affect the computational cost. When running on an i5 laptop with 12GB of RAM, the FS process consumed roughly 1-2% of the total system time. However, during the transmission of payoff matrices from the FC, CPU usage spiked to approximately 25%. On an i5 laptop with 8GB of RAM, the FC process utilized around 30% of the CPU during matrix exchanges.

4.2 Security

Our proposed method employs a robust multi-level authentication scheme. The initial phase involves a game-based challenge, a handshake process between the FC and FS. During this exchange, the value V is calculated using a Game Based Mixed Dominance Method. Even if an eavesdropper intercepts V, bypassing the subsequent authentication stages remains challenging. The second stage utilizes a one-way hash chain, $HASH^n(V_{FS} \,||\, K_{FS})$. The value V is repeatedly hashed n times, where n equals V. The inherent irreversibility of one-way hash functions enhances security. In the third stage, symmetric encryption is used to securely encrypt and decrypt messages and keys shared between FS and FC. To further fortify security, all communications and key exchanges occur over two separate channels, making it extremely difficult for an eavesdropper to compromise the system.

4.3 Communication Cost

The number of interactions between FC and FS are relatively less in this protocol than other schemes and there is no certificate exchange process. In this section we analyze the communication cost of the proposed protocol. The computation cost is given in Table 4 we can determine the communication complexity. Initially the client's metadata of size 128 bits is shared to the server. Matrix is shared between both participants before performing their subsequent communication of 246 bytes each. A crypto string of size 113 bytes is shared The FU-FS secret key K (FU) FS is known only to FU and FS, and is used to encrypt/decrypt for Ks. Therefore, the proposed scheme provides confidential communication.

4.4 Proof of Proposed Work

a) Theorem 1: In this authentication scheme, if the participating FC performs all the steps efficiently and completes the authentication process, the authenticating server FS always approves FC as valid.

Proof of Theorem 1: If the FC follows the exact procedure of the game challenge, dynamic payoff matrix is generated and matrices are exchanged between FS and FC using multi channels. This eliminates the necessity of securing the storage as everything is dynamically generated and authenticated. For the exchange, $FS \rightarrow FC$: $ENC\left(HASH^n(V_{FS} \,||\, (K_{FS})\right)$ is valid, and if $FC \rightarrow FS$: $ENC\left(HASH^n(V_{FC} \,||\, K_{FC}||T_{elapsed})\right)$ is also valid, then FS checks the match. If satisfied, the authentication process is complete and successful.

b) Theorem 2: It is impossible for a malicious user to access the keys as it is infeasible to access the matrix

conveyed through a predetermined channel and get itself authenticated.

Proof of Theorem 2: Consider a malicious user, M, attempting to deceive FS by hacking FC's metadata and claiming to be the legitimate user. FS communicates the challenge to FC through its established communication channel, which requires validation and authorization by FC itself. Assuming M successfully hacks FC's metadata and sends it to RS, the following occurs: i) T_i is transmitted through the client's Bluetooth channel. ii) T_i must be validated by C through the Wi-Fi channel. iii) If verification matches, the subsequent authentication steps continue through the same communication channel. Given the inherent security of the predetermined channel and the requirement for FC's validation, it is extremely challenging for a malicious user to intercept and manipulate the matrix, thus preventing unauthorized access to the keys.

4.5 Time Complexity

In this section, we analyze the efficiency of the proposed protocol in terms of Server Authentication time, Computation cost, Channel Verification time and Storage space for computation.

4.5.1. Authentication Time

In the proposed scheme, we use symmetric encryption for which we randomly generate two matrices: matrix $[A]$ at client side and matrix $[B]$ at server side. On a core i5 3rd gen processor with 12 GB RAM, these two matrices are exchanged across two different channels. The time taken to exchange matrices and to generate payoff matrix using bitwise XOR on the two matrices, say, $[payoff_{FS}] = [A] \oplus [B]$ on both sides is 0.072 sec, client and server. Time taken to obtain the value V by dominance method and generate Key, $K_{FS} = (MAC_i + BT_i) \oplus TS_i$ is 0.0752 sec. The FS with core i5- 4 gen @ 1.6 GHz, 12 GB RAM verifies the physical channel (Bluetooth) of FC with core i5- 11gen @ 3 GHz, 8 Gb RAM in 0.01 sec of time. The average server authentication time for hundred nodes is approximately 0.088sec as shown in Fig. 2.

TABLE III. Time complexity

Description	**Time in sec**
Time taken to generate matrix	0.001
Time taken to exchange two matrices and generate payoff matrix	0.072
Time taken to hash n times	0.075
Time taken to run entire code	0.0752
Channel verification time	0.01

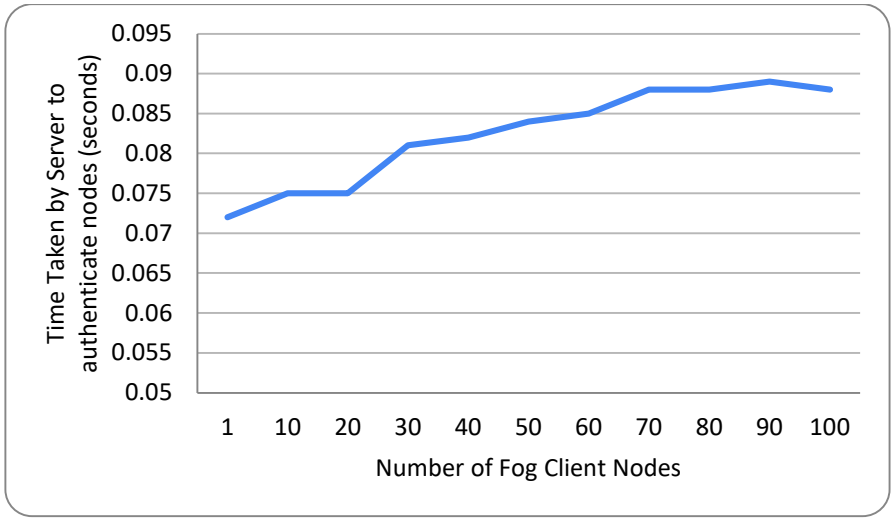

Fig. 2 Server Authentication Time

4.5.2. Computation Time

In this scheme FC takes about 0.075 sec (for n times SHA-256 hash invocations), 0.06 sec for one AES-128 block cipher encryption and 0.0752sec for decryption on the entire game challenge. So the overall client computation time is approximately 0.21sec. The FS takes about 0.75sec (for n times SHA-256 hash invocations), 0.06sec for one AES-128 block cipher encryption and 0.757sec for decryption on the entire game challenge. So the average computation time at the FS is 0.21sec.

4.6 Space Complexity

In our proposed method, the RS stores each FC's metadata of size 128 bits which includes 32 bits IP address IP_i, 48 bits MAC address MAC_i, 48 bits Bluetooth address BT_i in Blockchain. The total storage space required will be calculated as the storage per node multiplied by number of fog servers. For blockchain, a minimum of 4 nodes is typically required. Therefore, the total storage will be 4 times the storage per node.

5. Conclusion

We propose HMAM, a novel authentication mechanism designed to address the growing security challenges in IoT environments. HMAM leverages a matrix-based key exchange model and a two-channel communication approach to provide robust security and efficiency. To enhance the security, we use a multi-level authentication process, combined with the use of cryptographic techniques and decentralized storage, significantly strengthens security against various attacks. The HMAM is optimized for resource-constrained IoT devices, requiring minimal computational overhead. The use of multiple channels for communication minimizes latency and improves overall system responsiveness. The decentralized nature of HMAM allows for easy scalability to accommodate a growing number of IoT devices and fog nodes. In summary, HMAM offers a promising solution for securing IoT environments by providing a robust, efficient, and scalable authentication mechanism.

Conflict of Interest

The Author do not have any Conflict of Interest

References

1. Alzoubi, YI, Osmanaj, VH, Jaradat, A, Al-Ahmad, A. (2021). Fog computing security and privacy for the Internet of Things applications: State-of-the-art. *Security and Privacy*, Vol. 4(e145). https://doi.org/10.1002/spy2.145

2. Ashi, Zain & Al-Fawa'reh, Mohammad & Al-Fayoumi, Mustafa. (2020). Fog computing: security challenges

and countermeasures. International Journal of Computer Applications. 175. 30-36. DOI:10.5120/ijca2020920648
3. I. Butun, A. Sari and P. Österberg, (2019). "Security Implications of Fog Computing on the Internet of Things," 2019 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, pp. 1-6, DOI: 10.1109/ICCE.2019.8661909
4. Khan, S., Parkinson, S. & Qin, Y. Fog computing security: A Review of Current Applications and Security Solutions. *J Cloud Comp* 6, 19 (2017). https://doi.org/10.1186/s13677-017-0090-3
5. Singh, Parminder & Nayyar, Anand & Kaur, Avinash & Ghosh, Uttam. (2020). Blockchain and Fog Based Architecture for Internet of Everything in Smart Cities. Future Internet. 12. 1-12. DOI: 10.3390/fi12040061
6. H. P. Asha and I. Diana Jeba Jingle, (2022). "One Time Password-Based Two Channel Authentication Mechanism Using Blockchain," Data Science and Security, Lecture Notes in Networks and Systems, vol. 462, pp. 229–237. Springer, Singapore. https://doi.org/10.1007/978-981-19-2211-4_20
7. Murugesan, A, Saminathan, B, Al-Turjman, F, Kumar, RL. (2021). Analysis on homomorphic technique for data security in fog computing. Trans Emerging Tel Tech., Vol. 32 (e3990). https://doi.org/10.1002/ett.3990
8. S. Benadla, O. R. Merad-Boudia, S. M. Senouci and M. Lehsaini, (2022). "Detecting Sybil Attacks in Vehicular Fog Networks Using RSSI and Blockchain," in IEEE Transactions on Network and Service Management, vol. 19, no. 4, pp. 3919-3935, DOI: 10.1109/TNSM.2022.3216073.
9. Khan, S., Parkinson, S. & Qin, Y. (2017). Fog computing security: a review of current applications and security solutions. *J Cloud Comp* Vol. 6 (19). https://doi.org/10.1186/s13677-017-0090-3
10. P. Kumar, N. Zaidi and T. Choudhury, (2016). Fog computing: Common security issues and proposed countermeasures," 2016 International Conference System Modeling & Advancement in Research Trends (SMART), Moradabad, India, pp. 311-315, doi: 10.1109/SYSMART.2016.7894541.
11. Brief About The Challenges with Fog Computing| yourtechdiet https://yourtechdiet.com/blogs/fog-computing-issues/
12. Umoren, O.; Singh, R.; Pervez, Z.; Dahal, K. (2022). Securing Fog Computing with a Decentralized User Authentication Approach Based on Blockchain, Sensors, Vol. 22 (3956). https://doi.org/10.3390/s22103956
13. M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras and H. Janicke, (2019). Blockchain Technologies for the Internet of Things: Research Issues and Challenges," in IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2188-2204, DOI: 10.1109/JIOT.2018.2882794
14. Alam, Tanweer. (2020). Design a blockchain-based middleware layer in the Internet of Things Architecture. JOIV : International Journal on Informatics Visualization. 4. 10.30630/joiv.4.1.334
15. Khan, Neelam & Chishti, Mohammad Ahsan. (2020). Security Challenges in Fog and IoT, Blockchain Technology and Cell Tree Solutions: A Review. Scalable Computing: Practice and Experience. 21. 515-542. 10.12694/scpe.v21i3.1782
16. Alzoubi, Yehia & Al-Ahmad, Ahmad & Jaradat, Ashraf. (2021). Fog computing security and privacy issues, open challenges, and blockchain solution: An overview. International Journal of Electrical and Computer Engineering. 11. 5081-5088. 10.11591/ijece.v11i6.pp5081-5088
17. Wang, Qin & Zhu, Xinqi & Ni, Yiyang & Gu, Li & Zhu, Hongbo. (2019). Blockchain for the IoT and Industrial IoT: A Review. Internet of Things. 10. 100081. DOI: 10.1016/j.iot.2019.100081
18. Chiang, Mung & Ha, Sangtae & I., Chih-Lin & Risso, Fulvio & Zhang, Tao. (2017). Clarifying Fog Computing and Networking: 10 Questions and Answers. IEEE Communications Magazine. 55. 18-20. DOI:10.1109/MCOM.2017.7901470

19. Baniata, Hamza and Attila Kertész. “A Survey on Blockchain-Fog Integration Approaches.” IEEE Access 8 (2020): 102657-102668. DOI: 10.1109/ACCESS.2020.2999213
20. Patwary, A.A.N.; Fu, A.; Battula, S.K.; Naha, R.K.; Garg, S.; Mahanti, A. FogAuthChain: A secure location-based authentication scheme in fog computing environments using Blockchain. Comput. Commun. 2020, 162, 212–224. https://doi.org/10.1016/j.comcom.2020.08.021
21. Y. Imine, D. E. Kouicem, A. Bouabdallah and L. Ahmed, "MASFOG: An Efficient Mutual Authentication Scheme for Fog Computing Architecture," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, USA, 2018, pp. 608-613, doi: 10.1109/TrustCom/BigDataSE.2018.00091.
22. Pardeshi, M.S.; Sheu, R.-K.; Yuan, S.-M. Hash-Chain Fog/Edge: A Mode-Based Hash-Chain for Secured Mutual Authentication Protocol Using Zero-Knowledge Proofs in Fog/Edge. *Sensors* 2022, *22*, 607. https://doi.org/10.3390/s22020607
23. Jia, Xudong & Hu, Ning & Yin, Shi & Zhao, Yan & Zhang, Chi & Cheng, Xinda. (2020). A2 Chain: A Blockchain-Based Decentralized Authentication Scheme for 5G-Enabled IoT. Mobile Information Systems. 2020. 1-19. 10.1155/2020/8889192.
24. Baucas, Marc Jayson & Spachos, P. & Plataniotis, Konstantinos. (2021). Public Key Reinforced Blockchain Platform for Fog-IoT Network System Administration. IEEE Internet of Things Journal. PP. 1-1. 10.1109/JIOT.2021.3104740.
25. Liu, X., Chen, W., Xia, Y., & Shen, R. (2022). TRAMS: A Secure Vehicular Crowdsensing Scheme Based on Multi-Authority Attribute-Based Signature. *IEEE Transactions on Intelligent Transportation Systems, 23*, 12790-12800.
26. Khashan, Osama. (2020). Hybrid Lightweight Proxy Re-Encryption Scheme for Secure Fog-to-Things Environment. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.2984317.
27. Ibrahim, Maged. (2016). Octopus: An Edge-Fog Mutual Authentication Scheme. International Journal of Network Security. 18. 1089-1101.
28. Wei, Chao & Fadlullah, Zubair & Kato, Nei & Stojmenovic, Ivan. (2014). On Optimally Reducing Power Loss in Micro-grids With Power Storage Devices. Selected Areas in Communications, IEEE Journal on. 32. 1361-1370. 10.1109/JSAC.2014.2332077.
29. Diana Jeba Jingle, I., Mano Paul, P. (2021). A Fog-Based Retrieval of Real-Time Data for Health Applications. In: Bhattacharyya, S., Mršić, L., Brkljačić, M., Kureethara, J.V., Koeppen, M. (eds) Recent Trends in Signal and Image Processing. ISSIP 2020. Advances in Intelligent Systems and Computing, vol 1333. Springer, Singapore. https://doi.org/10.1007/978-981-33-6966-5_16.
30 Hilda Janice, J., Jingle I, D.J. (2021). An IoT-Based Fog Computing Approach for Retrieval of Patient Vitals. In: Jat, D.S., Shukla, S., Unal, A., Mishra, D.K. (eds) Data Science and Security. Lecture Notes in Networks and Systems, vol 132. Springer, Singapore. https://doi.org/10.1007/978-981-15-5309-7_18.
31. Jingle, Diana & Paul, Mano & Jayapalan, Daniel Francis Selvaraj. (2019). EShield: An Effective Detection and Mitigation of Flooding in DDoS Attacks over Large Scale Networks. 8. 1557-1562.
32. Paul, P.M., Shekhar, R., Jingle, I.D.J., Jingle, I.B.J. (2024). Prevention and Mitigation of Intrusion Using an Efficient Ensemble Classification in Fog Computing. In Proc. IC3T 2023, Lecture Notes in Networks and Systems, Vol. 898. Springer, Singapore. https://doi.org/10.1007/978-981-99-9707-7_16