Survey on IoT Device Authentication Protocols from Hash Based Schemes to Blockchain Based Schemes

Sanâ Elaoudi(✉), Marouane Sebgui, and Slimane Bah

Ecole Mohammadia d'Ingénieur, University Mohammed V, Av. Ibn Sina, Rabat, Morocco
sana-elaoudi@um5.ac.ma, {sebgui,slimane.bah}@emi.ac.ma

Abstract. The use of IoT devices especially sensors and actuators continue to emerge across many domains: Smart city, Smart vehicle, Smart healthcare, Smart Factory. . . and influences the way we live and act nowadays. Nevertheless, IoT networks are subject to many security concerns related to privacy, data integrity, traceability, reliability and more. Device authentication is the key stone to prevent security attacks, guarantee the efficiency of the IoT network and ensure a reliable information exchange. Many works have been done around device-to-device and user-to-device authentication protocols and different factors have been considered: RFI tags, MAC address, PUF, Unique ID, . . . all aiming to provide an efficient and secure authentication scheme that suits the heterogeneous resource constrained IoT devices. In this paper we discuss several ways of device-to-device authentication for IoT applications presented by researchers to ensure enhanced authentication protocols and key exchange mechanisms. We also highlight common challenges for these mechanisms in relation with timestamp accuracy and time synchronization which seems to be a green field for scientific researches related to IoT.

Keywords: IoT · device authentication · blockchain · timestamp · hash function · cryptography · PTP

1 Introduction

Internet of Thing is an emergent concept of connected devices that is applicable in every domain involving healthcare, agriculture and industry.

Indeed, IoT devices (sensors, actuators, ...) gather environmental data and distribute the real time information to facilitate, automate and accelerate the decisions making. E.g, by sensing the patient blood pressure, temperature and glycemic variants, the Internet of Medical Thing assists medical staff to prevent unexpected health troubles. [1]

Same approach is experienced when Industrial Internet of Thing devices serve manufacturers to automate the procurement and the products inventory. [2]

T. Pereira et al. (Eds.): IoECon 2024, LNICST 607, pp. 12–29, 2025.
https://doi.org/10.1007/978-3-031-84426-3_2

Furthermore, special types of sensors are used in agriculture to monitor temperature, irrigation and sun radiation for plants. [3] IoT devices may be connected through different network technologies: Zigbee, WIFI, Bluetooth, ... depending on the purpose and usage conditions. IoT Networks contain usually heterogenous and resource constrained entities in terms of computation capability, internal storage and battery lifetime. This induces IoT network to use different communication and authentication protocols. [4]

To cope with the challenging issues related to the heterogeneity and resource constraints of IoT networks, scientists are involved in a continuous research process to standardize the communication and authentication processes.

Interesting works have been conducted around IoT devices and applications with authentication as a dominant subject when talking about facing IoT Security challenges. Providing network access to a compromised IoT device can lead to severe security attack and compromise the entire network.

Since IoT devices have usually access to sensitive data related to one's health parameters, identity credentials, habits and environment variants, prohibition of unauthorized access to a device becomes a priority. [5]

Indeed, authentication is an imminent need for IoT network to cope with critical security aspects:

- Authentication ensures data integrity by tightening rights to data processing and delivery only to legitimate devices.
- Authentication enhances data confidentiality where only authenticated devices can access the network and get data.
- Non-repudiation is another advantage of authentication by making possible to retrieve device identity or pseudoidentity after authentication and link each action to its actor.
- Authentication guarantees the Data integrity while denying access to unauthenticated devices. [6]

These are not the only benefits of IoT device authentication. Referring to its standard definition, the authentication process consists of allowing access based on unique identity validation using a set of parameters securely allocated to a device. Therefore, authentication itself is an imperative part of Access Control and Management Systems. [7]

Over years, many authentication schemes have shown interest using different techniques.

One of these techniques is digital signatures and hash functions that have been widely used and consisted of exchanging Public/Private keys among devices to authenticate each other. [8] Considering these techniques, usually the authentication requires a central third-party entity like certificate server, network operator, a base station or an authentication server. The central entity stores device credentials and verifies the identity and access rights of the device submitting the authentication request. [13,14]

This is a centralized architecture that proved a high level of security but showed drawbacks like single point of failure when the central entity is compromised or fails to function. [17]

Digital signature-based authentication model can also be independent of any third-party entity and permit authentication among devices only with no need for a central entity. In this case, the architecture is decentralized avoiding single point of failure but induce additional costs for communication over a secure channel and moreover requirements for high level of computation and storage capabilities from the devices. [12]

Based on smart contracts, Blockchain has gained interest facing these weaknesses by automating and simplifying the authentication computation complexity and communication overhead. [9]

Initially, Blockchain has been introduced for cryptocurrency afterwards the use of Ethereum as an operational environment for developers and software designers incited a big convergence to Blockchain. Ethereum simplifies the creation and compilation of smart contract to automate action under triggers.

In respect to authentication schemes, Blockchain is used to build a decentralized model. Decentralization enables direct authentication requests among devices while the request validation or rejection is automatized through smart contract according to various parameters (identity, pseudo-identity, rights, roles, keys...etc.). [10]

Blockchain based authentication schemes are secured against Denial-of-Service attack and prevent malicious nodes from launching an attack due to Blockchain transactions expensive fees. [16]

Therefore, either it's an authentication using cryptography and hash function or an authentication through blockchain platform, timestamp of authentication request or transaction is a crucial parameter to verify the freshness of the demand and preserve the scheme from security attacks like Replay. [11]

Timestamp is also used in concatenation with other parameters (unique identifier, PKI, passwords...etc.) as a part of the authentication request messages and requires high attention concerning time synchronization, time retrieve and time variation. [16,17]

In this paper, we analyze several existing authentication protocols for IoT devices authentication. We compare schemes employing simple operations and techniques like hash function, XOR operation and concatenation to schemes using blockchain as the authentication environment for the IoT devices.

Main criteria to consider an article in this study are recent publication (2018 to 2024), article relevance to scientific advancement (conference type and number of cites), IoT device mutual authentication schemes only (user authentication schemes are excluded) and finally proposed scheme based only on hash function, XOR operation and blockchain (other techniques like PUF, finger print, ... are excluded).

The remainder of this article is organized as follows. Section 2 presents related work. Section 3 introduces the Cryptography and hash functions. Section 4. introduces Blockchain Technology and main principle. Section 5. Presents a comparison among authentication schemes and the result analysis. And Sect. 6. Concludes the article and gives a preview of the future work.

2 Related Works

Public/Private Key mechanism combined to Hash functions are widely used for authentication schemes to store and verify devices identities and sign digitally every data exchange session. [12] proposes a scheme based on a network topology including central entity that provide system parameter for further partial private keys and session keys computation. Each network entity authenticates to every other entity before data transmission using several hash function values and partial private and public keys combinations. [13] presents a simpler scheme based on devices MAC Addresses published by the Base Station and the Cluster Heads to local and public chain after an offline registration of any device that joins the Network. [14] introduces an authentication scheme using elliptic curve cryptography and device's Physical Unclonable Function. The device needs to perform data computation to respond to a challenge. To be authenticated, the device's challenge result mut be validated by the authentication server. In the scheme presented by [15], each device registers to a remote server using pre-shared PKIs. The authentication is accepted by validating authentication messages created using some hashing value of pre-shared keys combined with device identifies. In [16], a remote server stores credential for every device and gateway after an offline registration. The credentials consist of a pseudoidentity and a private key. A pairing phase is conducted between the remote server and the device before each authentication request to ensure the device is already registered to the remote server and the authentication request is accepted. After that, the pseudoidentity is refreshed and the new pseudoidentity is used for authentication.

Almost all the presented works are based on central remote server that stores, delivers and shares credentials through a secure channel. An offline registration phase to the remote server is a must to provide the credentials and ensure devices are recognized as legitimate to ask for authentication. In [17] blockchain is used as the authentication system for IoT devices to avoid network architecture centralization around one entity and dismiss the use of a separate secure channel to communicate credentials. Indeed, a smart contract is deployed on the blockchain to automate the access control through an access control list of legitimate users with permissions. The smart contract uses hash function, Ethereum address, public keys and devices ID to validate an authentication request. Another device authentication scheme that combines cloud, fog and blockchain technologies was proposed in [18]. The scheme considers three layers. First, the device layer for IoT devices that generate a pair of Private/Public key and request authentication to Access Managing Nodes (AMN) in the Fog layer. AMN are responsible for registering device by providing a device license that contains crucial information about the device: device unique identity, device public key, device's corresponding AMN unique identity and a signature using the corresponding AMN private key. The AMN forwards the device's details to the Blockchain which verifies the uniqueness of the device identity and stores the device's information for further use by the corresponding AMN. The cloud layer is solicited when IoT devices

from different domains need to communicate. Manager Nodes are designated to handle this communication among different AMN domains.

Either it's an authentication using cryptography and hash function or an authentication through blockchain platform, timestamp of authentication request or transaction is a crucial parameter to verify the freshness of the demand and preserve the scheme from security attacks like Replay. One essential entity for the timestamp is the time source. If this component is corrupted the whole authentication process is under fault. Consequently, a mechanism to ensure the time precision is a must for authentication.

Precision Time Protocol is an IEEE standardized protocol developed to synchronize clocks over a network or a network segment. The protocol architecture considers a Master-Slave model and three clock types. The ordinary clock also called the Grandmaster Clock is the time source while the Boundary Clock is connected to many network segments and has the role to synchronize these segments to each other. Finaly, the Transparent Clock, adds more precision to the synchronization messages by compensating the time spent for network traversal. As mentioned in [19], Precision Time Protocol succeeded to unify the time source among IoT devices but showed weakness against security attacks like modification, masquerading, delay, replay, and denial of service.

Many studies have been conducted around time synchronization for IoT devices, where blockchain [20] is used to ensure the time synchronization and prevent the tampering of the time broadcasted by the time source. Each time an IoT device needs to refresh the local time and ensure the synchronization with other network entity a transaction is initiated in the blockchain network. The transaction involves time source, which is the main time provider, consensus nodes that accept or refuse the transaction depending on the initiator authentication and the consensus protocol and time nodes that broadcast updated time to IoT devices at a special frequency.

3 Cryptography and Hash Functions

A significant number of IoT devices authentication schemes rely on a combination of cryptographic digital signatures and hashing algorithms to identify devices and authenticate their messages.

3.1 Cryptography

As its wide definition, [21] cryptography is the coding of an information in the form of a cipher text so only the intended receiver can decode it and read it clearly (encryption/decryption mechanism). This ensures enhanced data privacy and integrity.

Encryption and decryption require cryptographic keys that are strings of characters. When the same key is used for encryption and decryption, the cryptosystem is defined as symmetric e.g. Advanced Encryption Standard (AES) and

Data Encryption Standard (DES). While the algorithm is classified as asymmetric when different keys are used for encryption and decryption such as Elliptic Curve Cryptography. [22] Fig. 1

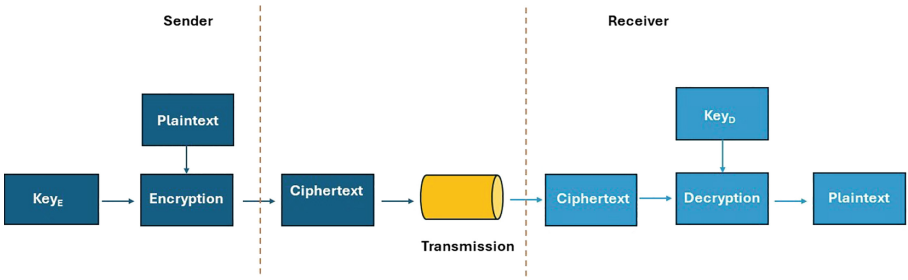

Fig. 1. Cryptography Process [23]

Symmetric algorithms are secure and fast but require pre-share of keys. The encrypted messages can be compromised if a malicious entity gets the keys.

On the other hand, asymmetric cryptosystems avoid the key sharing threats but require large key length which implies more convergence time and more difficulty. [24] In fact, asymmetric algorithms use different keys for encryption/decryption and generate heavy computation and communication overheads. This is the major drawback for the use of asymmetric cryptography for resource constrained IoT device authentication. [25]

Hence, authentication models based on cryptography mostly focus on symmetric algorithms to suit the limited resources of IoT devices using a unique key for encryption and decryption. [26] Nonetheless, introducing Elliptic Curve Cryptography made a great success employing a short key length that optimizes the requirement for computing capabilities and power consumption. [27]

Elliptic Curve Cryptography is, mathematically, a polynomial equation with a modulo reduction. The equation is represented in the binary field by an XOR operation. [27] gives a detailed description of the Elliptic Curve Cryptography operations.

3.2 Hash Function

Basically, Hash function are auxiliary functions in cryptography that are used for different purposes like digital signature, key derivation, Random Number Generation and Message Authentication Code (MAC). [28]

Hash functions are characterized by several properties:

- Arbitrary input length: Hash function accepts input from different sizes,
- Fixed output length: Hash function gives output in a specific predefined length,

- Pre-image Resistance also called one-way: given a hash value, it's almost impossible to find the corresponding input,
- Second Preimage Resistance or weak collision resistance: given an input, it's very difficult to find another input that corresponds to the same hash value.
- Collision Resistance: it's very hard to obtain two inputs that correspond to the same hash value. [29]

A hash function that satisfies these criteria is classified secure but if one of these properties shows a weakness, the hash function is faulty. [29]

Many hashing function were developed such as BLAKE, KECCAK, SHA, ... but only some of them have been standardized. Thus, a worldwide common principles among cryptography specialists is that secure hash functions have been standardized. New conceptions are not a focus, just existing standardized hash functions should be used or enhanced. [29] The most used hash function is the Secure Hashing Algorithm (SHA) that accepts a maximum input size of a 28 bits and gives an output of size 512 bits. SHA has been enhanced to SHA-1, SHA-2 and SHA-3 (KECCAK) to handle some vulnerabilities. [28]

SHA-3 represents a family of four cryptographic functions depending on the output size: SHA-224, SHA-256, SHA-384 and SHA-512. [30] Fig. 2

Hence, Hash function is an efficient way to sign a message in any size by a predefined size short signature. It's used to verify the integrity of an initial message. If any tampering happens during the transmission the hash value of the message changes consequently. [28]

Furthermore, hash function has been used to derive a set of several cryptographic keys from one initial key called the seed. This application of hash function is known as one-way-hash-function-chain that is very strong and cannot be inverted. [31]

Hash function represents also a vital concept for the Blockchain technology. Each blockchain block contains the hash value of the precedent block, therefore all the blocks are linked together to form a chain and any change even miniscule affect the whole chain. [32]

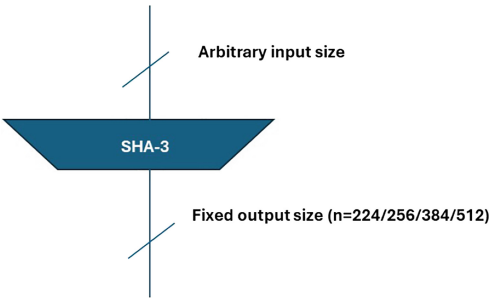

Fig. 2. SHA-3 cryptographic functions [30]

3.3 Digital Signature

Digital signature is a mathematical function that signs digitally a message by the sender's identity represented as a pair of Public/Private keys. [33] With Digital signature mechanism the message is signed by the sender's private key before transmitting and verified through sender's public key after its reception. This ensures the sender authentication and non-repudiation. On the other hand, digital signature guarantees the message integrity. In fact, in case the signed message has been altered by a malicious entity, the verification using public key process returns an error. [33] Fig. 3

When received, the signed message passes through a verification process. Using the sender's public key, the hash value of the initial message Mh' is retrieved. Then the cleartext initial message received is hashed by the same Hash function Mh.

The verification process consists of comparing Mh' and Mh: Fig. 4

- If Mh' and Mh are equal, the message is valid.
- If Mh' and Mh are differents, the message is invalid.

Authentication schemes can be classified over several domains including Public Key Cryptography and Digital Signature. [34]

Within the schemes based on digital signature, each device needs to choose a pair of public and private keys that uniquely identify it. Using this pair, devices can authenticate each other in two ways:

- Through a central third party that stores devices credentials. The central entity is solicited in every authentication request and has the role to verify devices credentials and authenticate them. This architecture is secured and efficient but presents the drawback of Single Point of Failure.
- Independently of any third-party entity by direct mutual authentication. This architecture avoids the single point of failure but requires computation and storage capabilities from devices [35]

After the authentication, the keys are used to secure the communication and are called the session keys. [26]

4 Blockchain Technology

Blockchain is a distributed ledger introduced initially by Nakamoto for financial transactions called cryptocurrency in bitcoin systems. Transactions are stored in a secure shared database over blocks. A block is a data structure of minimum 5 components: Fig. 5

- Hash of precedent block: the result of applying a specific hash function to the precedent block entire data,
- Timestamp: the current time of the block creation,
- Nonce: a unique value used once to generate the hash value of the block,

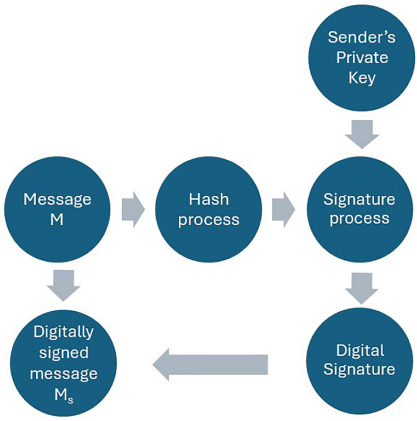

Fig. 3. Digital Signature Process [34]

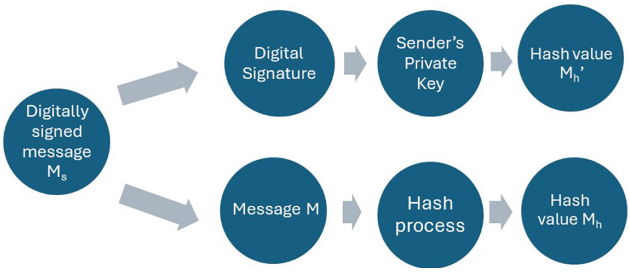

Fig. 4. Verification Process [34]

- Transactions: financial operations,
- MerkleRoot: concatenating the hash values of all the transactions in the block. Figure 6

The main principle of blockchain is that all the participants, also called nodes, must agree on a challenge (consensus) for a transaction to be accepted and stored into a block.

The consensus is an algorithm that incites blockchain nodes to proof their trustworthiness in a trustless environment.

There are numerous consensus algorithms: Proof of Work (PoW), Proof of Stake (PoS), practical byzantine fault tolerance (PBFT) ... but PoW is the commonly used consensus algorithm.

Under the PoW algorithm, the system imposes the complexity level criteria as a challenge. Participating nodes also called miners compete by computing a suitable nonce value. The challenge ends if a nonce value hashed with other block content conforms to the target value imposed by the system.

If no hash value meets the difficulty conditions, a new nonce is added and the calculation is reiterated.

The first node to generate a hash value that conforms to the challenge criteria is selected to add the transaction to a block. The elected node forwards the new block to other peers for verification.

Other nodes recalculate the block hash value and verifies if it agrees with the system criteria. [37]

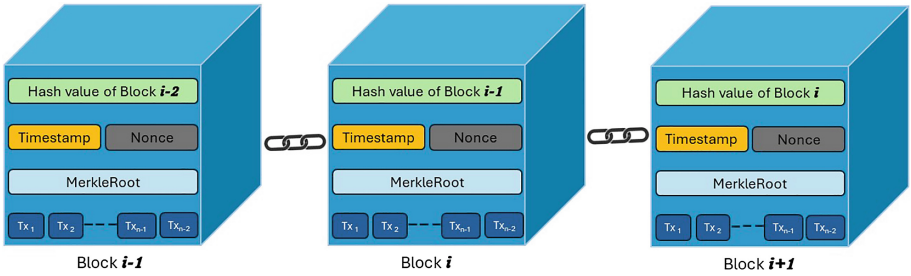

Fig. 5. Blockchain Block structure. [36]

To make it simple, blockchain is a chain of blocks each one containing an information part of the precedent blocks.

The immutability of the blocks is ensured by the hash value of the precedent block into the new block. Hence, all the blocks are linked together in a chain back to the first block (genesis) and the tampering of a single miniscule information in block data result in a completely different hash value. [38]

Nodes form a Blockchain Network by participating in the consensus algorithm and sharing Blockchain data. Therefore, nodes also called participant do not have the same role in the BC Network. In fact, a participant can be a full node or a light node:

- Full node: stores the entire blockchain data and verifies new blocks. A full node can be a miner that runs a computing software, to mine new transactions and satisfy the consensus if it owns very powerful computing capabilities.
- Light or partial node runs a light software client to enable access to the blockchain and issues transactions to the BC Network. Light node may only store a copy of block headers to verify transactions, relying on full nodes to provide any other necessary data. [38]

To be part of a BC Network, nodes must respect different operating rules and access modes depending on the blockchain type. There are many ways to classify Blockchain:

- Private Blockchain is considered if the devices need to be in a closed network (private) that can be only joined by preselected entities. E.g a private BC Network for a company only accessible to this organization departments in a isolated area.

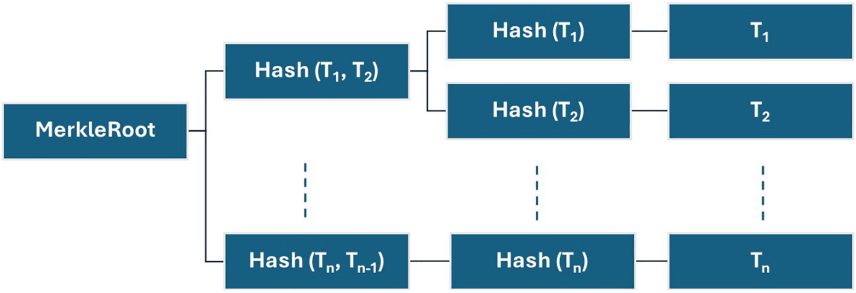

Fig. 6. MerkleRoot structure. [36]

- Public blockchain is used if the devices can operate in an open network (public) where anyone can access the network. E.g, Bitcoin Network characterized by a publicly open access.
- Permissioned blockchain: devices control is given to specific entities with a predefined permission,
- Permissionless blockchain, devices can be controlled by anyone in the network with no obligation for a permission. [39]

Besides, many blockchain platforms are available in the market and the choice to use one of these platforms is related to the purpose of the blockchain Network. Application designed for currency and transactions use transaction-oriented blockchains (bitcoin) whereas application designed to run a logic (smart contract) use logic-oriented blockchains [40].

Transaction-oriented Blockchain like bitcoin is mainly used to record permanently transactions performed by members. All bitcoin values transferred among participants are mined, verified and stored. [10]

Logic-oriented Blockchain like Ethereum is principally solicited by developers considered as a large distributed virtual machine for coding and running software programs. [40]

Several blockchain platforms exist like Ethereum, Hyperledger Fabric, and Corda although Ethereum is the most popular that provides a virtual machine space where smart contracts can be developed, tested and deployed as EVM bytecodes [41].

EVM bytecodes or simply Smart Contracts are stored software programs that run when special conditions are met. Smart Contract are time efficient and automate the execution of actions with no need for a third party. Smart contract transactions are transparent and shared among different nodes for traceability and immutability. [40]

Developers may use several developing languages like LLL (Low-level Lisp-like Language), Serpent (a Python-like language), Viper (a Python-like language), or Solidity (a Javascript-like language) but Solidity remains the key language for Ethereum [40]. As authentication protocols consist of software applications organizing and automizing the authentication process among IoT

devices/entities, blockchain has gain interest for securing and decentralizing authentication schemes. [42]

Various studies and schemes have been achieved around authentication protocols for IoT devices using blockchain platforms but requires more implementation costs, service fees and high capabilities for computation and storage. [43–45] Some works will be considered in Sect. 5 when comparing protocols for IoT device-to-device authentication.

5 Authentication Schemes Comparison

5.1 Comparison Results

In this study, we aim at highlighting the crucial key parameters to take in consideration while designing an authentication protocol for IoT devices.

On one hand, we have focused on schemes employing simple operations and techniques like hash function, XOR operation, concatenation...

On the other hand, we considered schemes using blockchain as the authentication environment for the IoT devices. Hence, we chose some existing schemes such as references [12,15,16,27,32,44,45] for comparison based on a list of common parameters.

Actually, the main criteria for evaluating an authentication protocol are:

- **Computation cost** that represents the computational size of operations to be performed to establish a successful authentication. In this paper, we consider hash function and XOR calculation as operations. We compare protocols based on the number of operations required to achieve a successful mutual authentication between two entities.

- **Communication overhead** that consists of the volume of message exchange to establish a successful authentication. Here, every communication between two entities while authenticating each other is counted as a message (Msg). We compare protocols based on the required information exchange for every authentication request/validation.

- **Efficiency in terms of time and latency** that ensure a fast protocol convergence. In our study, time is calculated based on the total duration required by all operations during the mutual authentication process:
  - TH is the time required for a Hash function,
  - TXOR is the time consumed for an XOR calculation,
  - TKeyGen is the time taken to generate a session Key,
  - Tsign is the time required for signing a message by a digital signature,
  - Tverif is the time needed to verify the sender's digital signature,

– **Protocol architecture** that can be centralized presenting the drawback of single-point-of-failure or distributed requiring advanced resources.
An architecture is classified as centralized when a central unit is the bridge for every mutual authentication between two entities.
the architecture is identified as decentralized when the units can reach each other and authenticate mutually with no need for a third party.

All involved works for the comparison contain phases for registration, paring, session-key generation, data transmission... Phases may be different for each single protocol, but the authentication phase is a common concern. Although in our comparison, we focus only on the authentication phase of each protocol. Table 1 and Table 2 represent a comparison of several existing authentication schemes based on the above parameters.

Table 1. Comparison of authentication schemes. Part 1

	C.Cost	C.Overhead	Efficiency
Reference [12]	8Hash + 4XOR	2Msg	8TH+4TXOR+2TKeyGen
Reference [15]	7Hash + 5XOR	2Msg	7TH+5TXOR+1TKeyGen
Reference [16]	12Hash + 3XOR	4Msg	12TH+3TXOR+3TKeyGen
Reference [27]	4XOR	3Msg	4TXOR+1TKeyGen
Reference [32]	1 Sign	1 Msg	1TSIGN+1Tverif
Reference [44]	1Sign	1Msg	1TSIGN + 1Tverif
Reference [45]	7Hash + 2Sign	3Msg	7TH+Tsign+2Tverif+2TKeyGen

C.Cost = Computation Cost, **C.Overhead** = Communication overhead, **Sign** = Digital Signature mechanism.

Table 2. Comparison of authentication schemes. Part 2

	Architecture Type	Reg.mode	Reg.method
Reference [12]	Decentralized	Offline	Secure channel
Reference [15]	Centralized	Offline	Secure channel
Reference [16]	Centralized	Offline	Open channel
Reference [27]	Centralized	Offline	Open channel
Reference [32]	Decentralized	Offline	Secure channel
Reference [44]	Centralized	Offline	Secure channel
Reference [45]	Decentralized	Offline	Secure channel

Reg.Mode=Registration Mode, **Reg.Method** = Registration Method

5.2 Analysis and Discussion

References [12,32,45] offer a decentralized architecture scheme to avoid single point of failure and accelerate authentication among devices. Thus, require an offline registration phase before any device can join the network. Offline registration limits the IoT network extensibility and delays devices authentication till the registration is completed.

References [15,44], in contrast, propose a centralized architecture that permits to any IoT device to join the network and initiate an authentication request at any time but requires a secure channel to exchange credentials. Extra costs are to be considered in this kind of architecture.

Furthermore, if the central unit is compromised, the whole authentication process fails.

Referring to references [16,27], when registration is conducted over an open channel, computation cost and communication overhead rise. This is due to the use of an important number of hash functions and XOR operations to secure the authentication phase from a malicious entity. On the other hand, IoT devices have limited resources, whereas the elevated computation and communication costs harden the authentication process.

Reference [27] represents the Lowest combination of computation cost and communication overhead but this doesn't mean that the proposed protocol is ideal. The protocol representing the best choice for authentication must have the best combination of all the criteria meaning that the protocol is fast converging, secured against common security attacks and least consuming of resources linked to computation and communication.

References [32,44] use blockchain as the authentication environment which reduce communication overhead and computation cost and fasten the authentication process. In contrast, [45] uses blockchain only for data storage. Authentication process is conducted directly between IoT devices which rises the communication overhead and computation cost and slows the authentication process compared to schemes proposed in [32,44].

Based on the comparison result we have noticed that a combination of several points should be taken in consideration to design a successful IoT authentication protocol. In summary, these are the most relevant aspect to respect when designing an authentication protocol:

- **A secure communication** is required when exchanging authentication details among IoT devices. Thus, two main methods can be used:

  - Exchange over a secure channel that prevents from the intrusion of a malicious entity,
  - Exchange over an open channel but with encrypted messages so malicious nodes cannot access authentication messages in plaintext.

- **A light computation capability** is imposed by the limited resources of IoT devices. So, the authentication process must be either:

– • Simple by using common calculation operations like XOR and Hash Function,
  • The heavy calculation should be ensured by a central unit like blockchain nodes.

– **A fast-converging authentication** process is a crucial parameter for the protocol efficiency. To fasten the protocol phases three main points should be ensured:

– • Use fast calculation operations,
  • Use the minimum number of calculation operations,
  • Minimize the messages exchanged between devices during authentication,

• **Choosing the authentication environment** influences the above criteria. Indeed, performing an authentication by involving IoT Network entities may imposes limitations concerning communication security, computation volume and time consumption. Although, operating an authentication over Blockchain Network changes the overall reasoning about security, limitations and constraints.

6 Conclusion

In this paper we looked at the main authentication methods for IoT devices; principally cryptographic based schemes and Blockchain based schemes. We also introduced the main principles and main components of each method. Furthermore, we evaluated several authentication schemes with respect to architecture type, computation cost and communication overhead parameters. We concluded that many research have been conducted around IoT authentication and achieved good results by giving birth to lightweight, secured and efficient authentication protocols. However, enhanced authentication protocol are a must in respect to secure communication, light computation cost, fast convergence and authentication environment.

On the other hand, since IoT networks are still growing, and IoT device resource constraints are appearing every single day more investigation in this field is a must.

We gave some tips based on our study to be used by other researches focusing in designing authentication protocols.

Our next work will focus on designing a lightweight authentication protocol that minimizes the IoT device effort and optimizes the authentication process. The main aspects highlighted in the result analysis paragraph represents our basis to conduct the future steps toward a successful completion.

References

1. Gatouillat, A., Badr, Y., Massot, B., Sejdic, E.: Internet of medical things: a review of recent contributions dealing with cyber-physical systems in medicine. IEEE Int. Things J. **5**(5) (2018)
2. Bartolomeu, P.C., Vieira, E., Hosseini, S.M., Ferreira, J.: Self-sovereign identity: use-cases, technologies, and challenges for industrial IoT. In: 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)
3. Palazzi, V., Gelati, F., Vaglioni, U., Alimenti, F., Mezzanotte, P., Roselli, L.: Leaf-compatible autonomous RFID-based wireless temperature sensors for precision agriculture, 2019 IEEE Topical Conference on Wireless Sensors and Sensor Networks (WiSNet) (2019)
4. Heeger, D., Plusquellic, J.: Analysis of IoT authentication over LoRa. In: 2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS) (2020)
5. Mazhar, N., Salleh, R., Zeeshan, M., Muzaffar Hameed, M.: Role of device identification and manufacturer usage description in IoT security: a survey. IEEE Access **9**
6. Kouicem, D.E., Bouabdallah, A., Lakhlef, H.: Internet of things security: a top-down survey. Comput. Netw. **141**(4), 199–221 (2018)
7. Omar, A.S., Basir, O.: Identity management in IoT networks using Blockchain and smart contracts. In: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (2018)
8. Lucia, O., Isong, B., Gasela, N., Abu-Mahfouz, A.M.: Device authentication schemes in IoT: a review. In: 2019 International Multidisciplinary Information Technology and Engineering Conference (IMITEC) (2019)
9. Kumar, N.M., Mallick, P.K.: Blockchain technology for security issues and challenges in IoT. Procedia Comput. Sci. **132**, 1815–1823 (2018)
10. Zhu, X., Badr, Y.: A survey on Blockchain-based identity management systems for the internet of things. In: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (2018)
11. Challa, S., et al.: Secure signature-based authenticated key establishment scheme for future IoT applications. IEEE Access **5** (2017)
12. Fang, D., Qian, Y., Hu, R.Q.: A flexible and efficient authentication and secure data transmission scheme for IoT applications. IEEE Internet Things J. (2020)
13. Adil, M., et al.: Hash-MAC-DSDV: mutual authentication for intelligent IoT-based cyber–physical systems. IEEE Internet Things J. (2021)
14. Alizai, Z.A., Tareen, N.F.: Improved IoT device authentication scheme using device capability and digital signatures. In: International Conference on Applied and Engineering Mathematics (2018)
15. Fadi A.L.T., Deebak, B.D.: Seamless authentication for IoT-big data technologies in smart industrial application systems. IEEE Trans. Ind. Inf. **17**(4) (2021)
16. Jan, M.A., Khan, F., Mastorakis, S., Adil, M., Akbar, A., Stergiou, N.: LightIoT-lightweight and secure communication for energy-efficient IoT in health informatics. IEEE Trans. Green Commun. Netw. **5**(3) (2021)

17. Panda, S.S., Satapathy, U., Mohanta, B.K., Jena, D., Gountia, D.: A Blockchain based decentralized authentication framework for resource constrained IOT devices. In: 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (2019)
18. Panda, S.S., Jena, D., Mohanta, B.K., Ramasubbareddy, B.D., Danes, M.: Authentication and key management in distributed IoT using Blockchain technology. IEEE Internet Things J. **8**(16) (2021)
19. Ferng, H.-W., Poernomo, M., Li, M.: An authentication scheme designed for the lightweight precision time protocol in IoT. In: 2020 IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech) (2020)
20. Fan, K., et al.: Blockchain-based secure time protection scheme in IoT. IEEE Internet Things J. **6**(3) (2019)
21. Hammi, B., Fayad, A., Khatoun, R., Zeadally, S., Begriche, Y.: A lightweight ECC-based authentication scheme for internet of things (IoT). IEEE Syst. J. **14**(3) (2020)
22. Gabsi, S., Kortli, Y., Beroulle, V., Kieffer, Y., Alasiry, A., Hamdi, B.: Novel ECC-based RFID mutual authentication protocol for emerging IoT applications. IEEE Access **9** (2021)
23. Sarkar, A., Chatterjee, S.R., Chakraborty, M.: The essence of network security: an end-to-end panorama. In: Chapter Role of Cryptography in Network Security. Springer (2020)
24. Bhardwaj, I., Kumar, A., Bansal, M.: A review on lightweight cryptography algorithms for data security and authentication in IoTs. In: 2017 4th International Conference on Signal Processing, Computing and Control (ISPCC)
25. Sridhar, S., Smys, S.: Intelligent security framework for IoT devices cryptography based end-to-end security architecture. In: 2017 International Conference on Inventive Systems and Control (ICISC) (2017)
26. Bettoumi, B., Bouallegue, R.: Evaluation of authentication based elliptic curve cryptography in wireless sensor networks in IoT context. In: 2018 26th International Conference on Software, Telecommunications and Computer Networks (SoftCOM) (2018)
27. Tewari, A., Gupta, B.B.: A mutual authentication protocol for iot devices using elliptic curve cryptography. In: 2018 8th International Conference on Cloud Computing, Data Science and Engineering (Confluence) (2018)
28. Rao, V., Prema, K.V.: Comparative study of lightweight hashing functions for resource constrained devices of IoT. In: 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS) (2019)
29. Mudler, V., Mermoud, A., Lenders, V., Tellenbach, B.: Trends in Data Protection and Encryption Technologies. Springer (2023)
30. Wu, X., Li, S.: High throughput design and implementation of SHA-3 hash algorithm. In: 2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC)
31. Dinh, T.T.A., Liu, R., Zhang, M., Chen, G., Ooi, B.C., Wang, J.: Untangling Blockchain: a data processing view of blockchain systems. IEEE Trans. Knowl. Data Eng. **30**(7) (2018)

32. Li, D., Peng, W., Deng, W., Gai, F.: A Blockchain-based authentication and security mechanism for IoT. In: 2018 27th International Conference on Computer Communication and Networks (ICCCN) (2018)
33. Careja, A.-C., Tapus, N.: Digital identity using blockchain technology. Procedia Comput. Sci. **221**, 1074–1082 (2023)
34. Mughal, M.A, Luo, X., Ullah, A., Ullah, S., Mahmood, Z.: A lightweight digital signature based security scheme for human-centered internet of things. IEEE Access **6** (2018)
35. Albalawi, A., Almrshed, A., Badhib, A., Alshehri, S.: A survey on the authentication techniques in internet of things. In: 2019 International Conference on Computer and Information Sciences (ICCIS) (2019)
36. Dai, H.-N., Zheng, Z., Zhang, Y.: Blockchain for internet of things: a survey. IEEE Internet Things J. **6**(5) (2019)
37. Gemeliarana, I.G.A.K., Sari, R.F.: Evaluation of proof of work (POW) blockchains security network on selfish mining. In: 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI) (2018)
38. Ali, M.S., Vecchio, M., Pincheira, M., Dolui, K., Antonelli, F., Rehmani, M.H.: Applications of blockchains in the internet of things: a comprehensive survey. IEEE Commun. Surv. Tutorials **21**(2) (2019)
39. Fernández-Caramés, T.M., Fraga-Lamas, P.: A review on the use of blockchain for the internet of things. IEEE Access **6** (2018)
40. Wohrer, M., Zdun, U.: Smart contracts: security patterns in the Ethereum ecosystem and solidity. In: 2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE) (2018)
41. Ashizawa, N., Yanai, N., Cruz, J.P., Okamura, S.: Eth2Vec: learning contract-wide code representations for vulnerability detection on Ethereum smart contracts. Blockchain Res. Appl. **3**(4), 100101 (2022)
42. Zhaofeng, M., Jialin, M., Jihui, W., Zhiguang, S.: Blockchain-based decentralized authentication modeling scheme in edge and IoT environment. IEEE Internet Things J. **8**(4) (2021)
43. Mudhar, J.K., Kalra, S., Malhotra, J.: An efficient blockchain based authentication scheme to secure fog enabled IoT devices. In: 2020 Indo – Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN) (2020)
44. Rashid, M.A., Pajooh, H.H.: A security framework for IoT authentication and authorization based on blockchain technology. In: 2019 18th IEEE International Conference On Trust, Security And Privacy in Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE) (2019)
45. Vangala, A., Sutrala, A.K., Das, A.K., Jo, M.: Smart contract-based blockchain-envisioned authentication scheme for smart farming. IEEE Internet Things J. **8**(13) (2021)