SN Computer Science manuscript No.
(will be inserted by the editor)

FlexiChain: A Minerless Scalable Next Generation Blockchain for Rapid Data and Device Security in Large Scale Complex Cyber-Physical Systems

Ahmad J. Alkhodair · Saraju P. Mohanty ⋆ · Elias Kougianos

Received: 22 Jan 2022 / Accepted: 08 Apr 2022

Abstract The advancement of technology in several fields has given the opportunity to change the conventional ways of the daily services and activities to a better, easier, efficient, and fully or partially automated manner. Cyber Physical System (CPS) application interactions using Distributed Ledger Technology (DLT) will enrich the conventional ways. We propose a DLT designed based on CPS application requirements to transform the traditional paradigm to a decentralized version. The proposed technology ensures the security of the process and the integrity of the participant in a flexible ledger that includes a virtual version of the actual nodes that are part of the network. The whole technology comprises of two essential algorithms: the registration and the authentication, each of which has been experimented with and analyzed. The experiment conducted for three different cases of 10, 20, and 30 nodes, respectively, and the average registration time was 0.48, 0.54, and 0.7 ms. The average authentication time for the three cases was 3, 2.42, 1.23 ms/tx.

Keywords Cyber-Physical Systems (CPS) · Internet of Things (IoT) · Distributed Ledger Technology (DLT) · Blockchain · Directed Acyclic Graph (DAG) · Data Security · Device Security · Sensor Security

Ahmad J. Alkhodair
Dept. of Computer Science and Engineering
University of North Texas, Denton, Texas, USA
E-mail: AhmadAlkhodair@my.unt.edu

Saraju P. Mohanty (Corresponding Author)
Dept. of Computer Science and Engineering
University of North Texas, Denton, Texas, USA
E-mail: Saraju.Mohanty@unt.edu

Elias Kougianos
Dept. of Electrical Engineering
University of North Texas, Denton, Texas, USA
E-mail: Elias.Kougianos@unt.edu

1 Introduction

Cyber Physical Systems (CPS) is an Internet of Things (IoT) integrated with actuators that can be controlled and make physical changes according to decisions and feedback [9]. Reliability, sustainability, cost, security, privacy, production, and maintenance are contributions behind altering the conventional interaction way to the physical world by CPS [16]. However, the exponential growth of "things" will produce huge challenges. The system's automation, sensing, interaction, analysis, storage, and security need to be maintained in a robust and concurrent way with the network expansion which will be problematic if done through a central authority [1]. In addition, communication redundancy will increase the centralization issue by decreasing the security and accuracy while increasing latency and power consumption [1].

Several solutions have been proposed to resolve the drawbacks of network expansion [28]. One is Distributed Ledger Technologies (DLT) due to their unique characteristics. DLTs are linked lists formed as a graph with one or more transactions generated by public users and validated by a miner (central authority) through a complex computational process or probability algorithm [22]. For example, Blockchain technology has drawn a lot of attention since the launch of Bitcoin. However, the blockchain technology is mainly built for a cryptocurrency and requires enormous computational power. In addition, to add a block, the previous block must be appended to the chain first. Thus, the blockchain technology will not replace or be employed to resolve CPS challenges since it has been built to consume vast processing power [7]. Another example is tangle technology which was introduced to replace the heavy fees and fork issues in the blockchain technology by using a Directed Acyclic

Graph (DAG) as a linked list and to exclude the rule of miners. Even though it succeeds to solve the fork issue and minimize the validation time, it is using the same process to reach consensus among users which is the Proof of Work (PoW) used to reach consensus on the confirmed transactions [12] and is using a coordination node to protect the network from attacks such as parasite chains. Since the technology still consumes huge processing power, it will not contribute to CPS.

In this paper, a new distributed ledger technology is proposed for CPS that fits the requirements of CPS (as listed in figure 1) and decentralizes the whole network by eliminating the role of miners, reduces the processing power by using lightweight algorithms, and increases security by using the right hash function that could accelerate the process and ensure its efficiency [1, 20]. The paper proposes a novel architecture by combining the conventional blockchain, as shown in figure 2.(a) with a DAG structure, as shown in figure 2.(b) to avoid forking. The proposed linked list could integrate multiple blockchains into a "FlexiChain" by ensuring their distinctions as shown in figure 2. The distinction illustrated in the figure uses different color for each blockchain that integrates to three others forming the FlexiChain. In addition, the technology will secure the actual devices within the network by creating a virtual copy of the nodes called Virtual Node (VN), published and part of FlexiChain as a block to ensure the integrity of nodes. The virtual copy will include a unique identity derived from the actual nodes' manufacturer's specifications [2]. The mirrored version is part of the FlexiChain and the first node represents the genesis block of the FlexiChain, as shown in figure 2.

The paper is organized as follows: Section 2 summarizes the novel contributions of this paper. Section 3 presents previous related works and shows a comparative perspective of FlexiChain compared to established research. Section 4 presents the proposed FlexiChain and consensus algorithm associated with it. Section 5 provides experimental results. Finally, Sections 6 and 7 conclude the paper and present directions for future research.

2 Novel Contributions

2.1 Addressed Problem

DLTs have been targeted for many applications as a potential way to enhance the performance in CPS. Current DLTs could possibly overcome some of the challenges. However, the recent DLTs still lack CPS requirements (as listed in figure 1) that suit constrained applications such as time, power consumption, and scalability requirements which will be an obstacle for a full application.

2.2 Novel Solution Proposed In The Current Paper

Minerless FlexiChain technology proposed in this paper is a platform for IoT and CPS applications. FlexiChain has been explicitly designed to deploy DLTs in CPS and the IoT and to transform their operations to Peer to Peer (P2P) instead of a centralized paradigm, as shown in figure 2. All the requirements in CPS and IoT applications have been considered during the design steps such as the limited processing power, real time response, scalability, device integrity, and security as listed in figure 1.

2.3 Minerless FlexiChain

In this paper, FlexiChain provides a minerless automated registration process and minerless authentication operations. Device integrity and security are targeted by establishing a node's mirroring ledger that is integrated with FlexiChain. FlexiChain has many unique characteristics such as the integration between conventional blockchain and DAG, as presented in figure 2.(c) and figure 2.(d).

The novel contribution of this paper is as follow:

1. To the best of our knowledge, this is the first technology that uses a device manufacturer's specifications to create a digital unique signature for each node stored within the chain.
2. To the best of our knowledge, this is the first technology using blocks and arcs of different types to identify a blockchain path direction.
3. The proposed technology uses a hardware trusted module for registration process purposes.
4. Moreover, this is the first technology that is built specifically for CPS and IoT by design to ensure the integrity of nodes acting within the network.
5. This technology uses distinct blocks for certain purposes as part of the whole network.
6. In addition, compared to other technologies, and to the best of our knowledge, it is the only fully decentralized technology with no miners, validators, or coordinators.

In the current paper, FlexiChain presents one blockchain, which is comprised of Virtual Nodes (VN) and uses distinct blocks and arcs. The rest of the blocks are nodes' exchanged cryptographic assets. Figure 3 shows how the VN blockchain is used and integrated in FlexiChain.

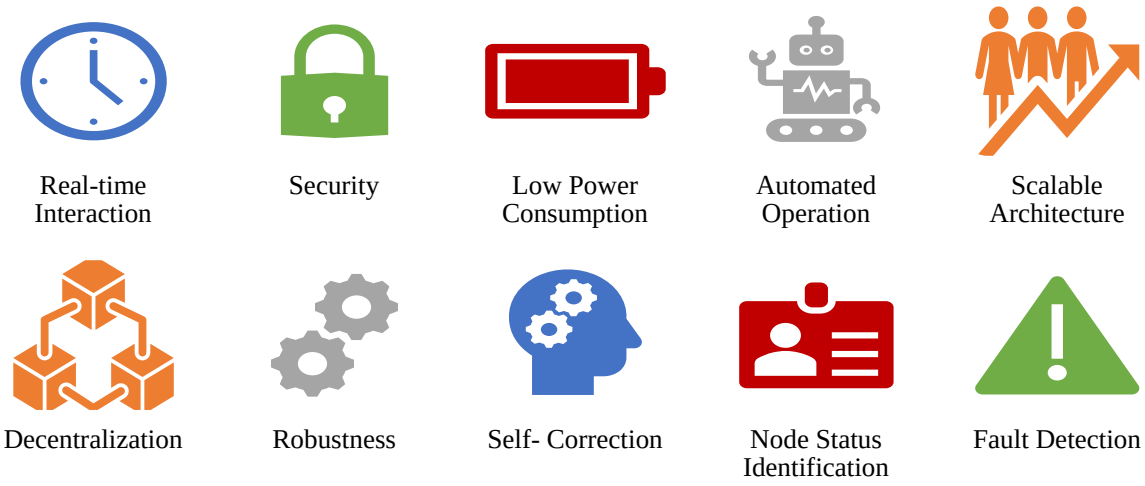


Fig. 1 Decentralized Cyber Physical Systems Features

3 Background and Prior Related Works

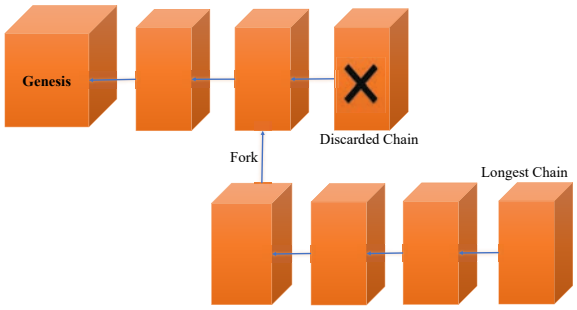
FlexiChain technology is designed for CPS and IoT environments based on their requirements. Decentralized Intelligent Transportation (DIT), Decentralized Smart Healthcare (DSHC), and Decentralized Supply Chain Management (DSCM) are examples of decentralized applications that are targeted in this technology since it takes into account resource capabilities and its integrity. The combination of conventional blockchain technology and DAG linked lists taking into account node capabilities and targeted application requirements by design will present the correct use of distributed ledger technology in CPS, and IoT applications. Figure 1 lists the requirements of CPS or IoT applications and to use DLT, these requirements should be satisfied.

3.1 Established Prior Related works

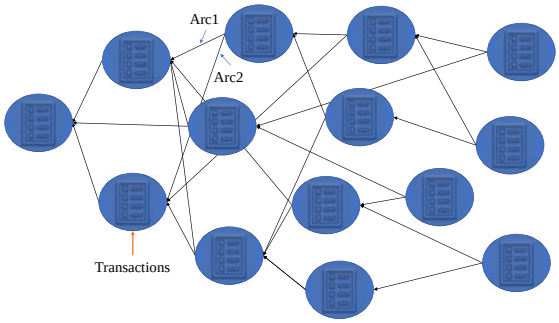
Blockchain technology is revolutionary and has shown efficiency in the Bitcoin and Ethereum cases. However, these two examples are public, decentralized, secure and they are not open for different use cases by design [9]. Building on top of them applications relying on limited capability nodes such as CPS and IoT applications will not be efficient. The linked list of the blockchain technology is organized and appended in order, which means no new blocks can be added until the current block is added or discarded from the network [14,18] In Bitcoin and Ethereum, the lack of scalability is a serious issue. In addition, blockchain technology deploys PoW the most energy consuming consensus algorithm, which ensures the unsuitability of this conventional paradigm in CPS, and IoT [18].

Several technologies have been proposed to resolve blockchain technology issues and its inability to fit IoT and CPS. Tangle technology has been the best as a replacement for the blockchain in the IoT and as a micro-payment system [12] that uses DAGs and single transactions as the data architecture [13, 27, 8, 31]. It resolves the issues of forks, miners, and huge fees. However, this technology is not suitable for resource-constrained devices due to the operations required. IOTA uses PoW as a consensus algorithm. It is well known that this mechanism requires massive calculations that might result in latency issues [12]. Several operations involved in IOTA could increase processing operations in addition to PoW, such as heavy selection algorithm. Tangle uses a coordination node that acts as a double checking node to ensure transaction validity, which indicate a weakness in its initial stages. By design, tangle technology has been structured for IOTA and has been built as an IoT micro-system. This is the only known use case, to the best of our knowledge.

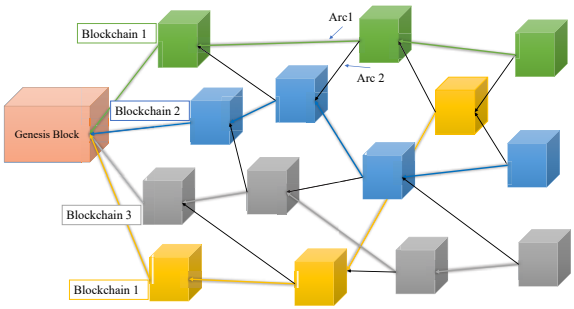
Hashgraph technology has has became a competitor due to its unique structure and gossip about gossip protocol [3, 4]. The technology uses containers of transaction hashes related by the voting system deployed. It requires high-resource nodes to operate efficiently. The throughput is very high but with high energy consumption due



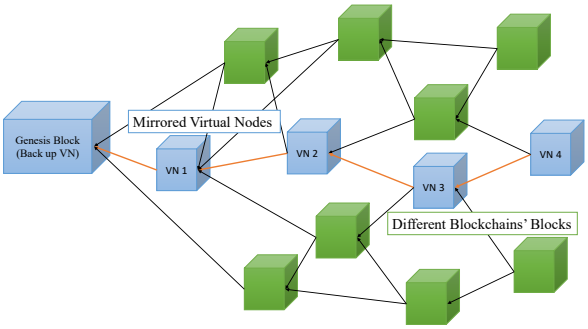
(a) Blockchain Conventional Linked List With Forking Issue Illustration



(b) Directed Acyclic Graph (Tangle Technology Scenario) Depection



(c) Proposed High-Level Depiction Of FlexiChain With Various Color Integrated and Interoperated Blockchains



(d) Current Paper Implementation High-Level Illustration Of FlexiChain With NodeChain [2] Integrated and Interoprated

Fig. 2 FlexiChain Design From Conventional Architecture Up to The Final Form Deception:Combination of DAG and Conventional Linked List

to the redundancy of the protocol. Hedera is the only real deployment of the hashgraph technology protocol.

Black-Lattice Technology uses a DAG architecture for the data linked list and PoW and Delegated Proof of Stake (DPoS) as consensus algorithms. The uniqueness of this technology is the chain of each node. This technology is targeting financial markets by using the most consuming protocols and requires high-resource nodes to function properly [15]. By design, this technology is unsuitable for CPS IoT applications.

Since blockchain technology is the most widely used version of DLTs, it has been the architecture for multiple cryptographic assets. In [30], a new customized conventional blockchain to fit IoT applications is presented. The type of blockchain is consortium type. Moreover, a combination of DPoS, Practical Byzantine Fault Tolerance (pBFT), and Verifiable Random Functions to produce "Roll-DPoS" that could fit IoT application requirements [30, 26]. Table 1 lists and compares some of the established related works to the proposed paper.

3.2 Research Prior Related works

The integration of DLTs, CPS and the IoT is very widespread among researchers and companies [32]. This research trend takes place due to advantages and uniqueness of DLTs [25]. Furthermore, observing the changes in recent technologies such as data architecture and consensus algorithms indicates that the conventional blockchain technology linked list causes scalability problems and the most popular protocols related to the consensus involve massive calculations or have huge redundancy, both consuming power and time [22]. Awareness of this necessity in custom designs for DLTs and within CPS and IoT requirements is shown in figure 1.

In [17] the proposed system is designed targeting security and speed. Three steps are associated with its operation: first, device enrollment is the process of enrolling new nodes to the network by generating responses from the challenges produced by the new device's Physical Unclonable Function (PUF) module [17]. The responses should satisfy certain requirements to enroll the new device to the network. The second step is the transaction initiation which is the process of broadcasting a transaction to the network, when a node collects data and the PUF module will generate responses and hash the data. Finally, the data will be broadcast to the network. The third step is the authentication step: the trusted node will receive the data and retrieve the predefined PUF keys from the secure database. If they matche, the block will be appended to the chain. If not, the process will be repeated [17]. In [11] the consensus protocol within the conventional architecture is built

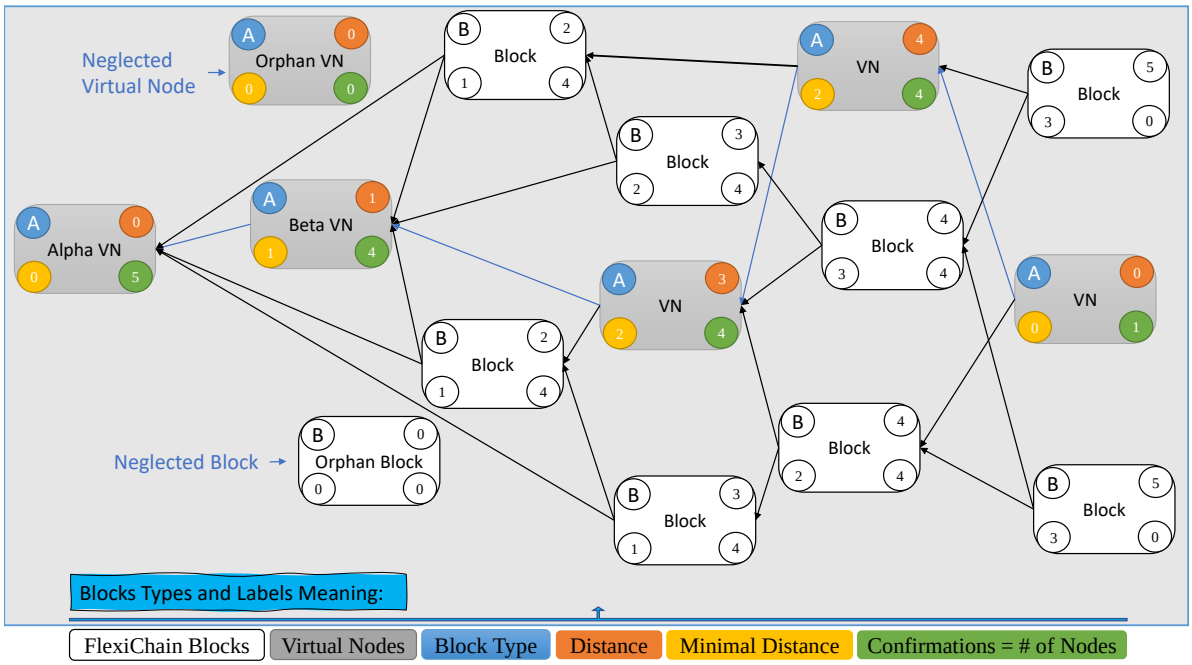


Fig. 3 FlexiChain Characteristics

to satisfy hardware security and speed authentication. Proof of Authenticity (PoA) is a protocol that ensures the protection of hardware and software aspects of a blockchain based IoT or CPS. The protocol functions in two levels: edge level and end-embedded sensor device level. This protocol uses an SRAM PUF module at the edge level to generate a public ID (challenge of devices) and private ID (responses of devices) and all the private keys are hashed to keep them secured. All the edge units are predefined to the network by the manufacturers. It involves two stages: edge registration stage and authentication stage [11]. In [23,24] a scalable consensus mechanism (PoAh) for device to device applications has been proposed. The proposed protocol is a lightweight version of the well-known proof of work. However, this protocol is faster and applicable to IoT environments. Since the nonce calculation in the traditional PoW is the time-consuming part, in PoAh there are no nonce calculation needed. Instead, the protocol depends on pre-defined addresses for miners and users. Thus, PoAh is suitable for a private blockchain. If a miner wants to authenticate a transaction, the miner should compare the MAC address of the sender to the one previously recorded in the database. Moreover, a regular node could transform to a miner node over time by having a certain number of authenticated transactions [23]. A protocol built for business blockchain based IoT, targeting scalability and security is presented in [6]. The protocol consists of two stages: the trade verification, and the consensus formation. Trade verification is the process of verifying the devices trading using the smart contract as a permission to trade. The consensus formation is the stage performed by the verification to reach an agreement on blocks that have been verified, in a limited time [6]. A recent works [10] proposes Distributed Smart Healthcare system (DSH) using smart contracts and a unique design to fulfill CPS requirements. The works consists of two major parts: decentralized operations and a distributed ledger through a distinct design to suit DSH. An access management system is integrated with the system to ensure security and integrity of resource constrained nodes. Another research work proposed in [5] uses a global blockchain for interoperability that could be used during a world crisis and uses updated and shared ledgers. Table 2 lists and compares some of the previous related research works to the proposed paper.

4 The Proposed Novel FlexiChain

Several DLTs in the market and used for Decentralized aaplications (Dapps) such as DeFi, NFT, and blockchain Interaction. However, none of them targets the integration of DLT and CPS and IoT. This creates some obstacles for employing the technology in CPS applications since it relies mainly in constrained resources devices. By design, FlexiChain targets CPS and IoT applications by using the lowest paradigm of computation and ensuring the system security and integrity.

FlexiChain technology is the integration of multiple conventional blockchains using a DAG structure and

Table 1 A Comparative Perspective of Blockchain, Tangle, Hashgraph and Minerless FlexiChain

Features	Blockchain Technology (for Bitcoin) [22, 18]	Tangle Technology (for Cryptocurrency) [12, 21]	HashGraph Distributed Ledger Technology [3, 4]	**McPoRa (Our Previous paper)** [1]	**Minerless FlexiChain Technology (current paper)**
Linked Lists	– Linked list of blocks – Each block contains multiple transactions	– DAG linked list – One transaction	– DAG linked List – Container of transaction hash	– DAG linked List – Each block contains multiple transactions	– Genesis Blockchain (independent ledger) – DAG linked list
Registration	Manual	Manual	Manual	Manual	Pre-Installed or Equipped Manufacturer Trusted Modules
Type of Validation	Mining	Mining	Virtual voting (witness)	Authentication (Minerless)	Authentication (Minerless)
Validators	Miners	Transactions	Containers	All Nodes	All Virtual Nodes
Types of Nodes	– Traders – Miners	– Traders – Coordinators	– Users	– Users	– Users – Back up
Number of Chains	One Chain	One Chain	One Chain	Multi-Chain	Multi-Chain: An Identified and Integrated NodeChain
Cryptography	Digital Signatures	Quantum key signature	Digital Signatures	Digital Signatures	– Trusted modules keys – Post-Constructed Digital Signatures
Hash Function	SHA 256	KECCAK-384	SHA 384	SCRYPT	SCRYPT
Consensus	Proof of Work	Proof of Work	Asynchronous Byzantine Fault Tolerance (ABFT)	Predefined UID Authentication	Two Factor Authentication: Constructed Public ID and Constructed UID
Numeric System	Binary	Trinity	Binary	Binary	Binary
Energy Requirements	High	High	Medium	Low	Low
Node Requirements	High Resources Node	High Resources Node	High Resources Node	Limited Resources Node	Limited Resources Node
Design Purpose	Cryptocurrency	IoT Cryptocurrency	Cryptocurrency	IoT/CPS Applications	IoT/CPS Applications
Block type	One	One	One	One	Two Blocks: – MC Block – VN Block – more as needed.

is explicitly built for CPS applications. The flexibility of the linked list proposed introduces a new way to build distributed ledger in a CPS environment due to its uniqueness. For example, combining several blockchains in FlexiChain by creating unique blocks and transactions types. Each transaction's type is gathered in the network pool and is accumulated in the same block's type. Each block arc and type represent a certain blockchain within the FlexiChain. The arcs of blocks are indicators to which blocks should be attached.

In the proposed paper, FlexiChain uses DAG and its arcs to define one blockchain as well as block type to keep the registration and authentication operations robust. By setting one arc for VN representation, the blockchain is based on a random filtration algorithm. Figure 3 represents some of the proposed characteristics, such as block and arc type, destination, confirmation, and how the whole architecture can avoid malicious VN and blocks. In the same figure, the growth of VN is not affecting the growth of FlexiChain since the blocks and arcs are recognized. Algorithm 1 is the process of the main operation and how the system will observe transaction type and which process it should follow.

A ledger is built to secure the nodes exchanging digital assets while securing the nodes and ensuring their integrity. The blocks in FlexiChain are connected to two previous blocks. One of the arcs will be connected to the same type of block and the other arc will be connected randomly. The FlexiChain linked list grows in a topological order as a whole graph presenting a strongly connected linked list. FlexiChain technology includes the integrity of the resource-constrained devices with its features by creating virtual copies within the chain using a certain type of block to store their Unique Identities

Table 2 A Comparative Perspective of Minerless FlexiChain with Previous Works

Consensus Algorithm	Registration (ms)	Authentication (ms)	Ledger	Miners	Validation	Blockchain Type	Linked List
Proof of Importance (PoI) [19]	Manual	60,000	Full	Yes	Accounts Importance	Public	Blockchain
Proof of Authority (PoA) [29]	Manual	5000	Full	Yes	PoS	Permissioned	Blockchain
Proof of Authentication (PoAh) [24]	Manual	3000	Full	Yes	Cryptographic	Private	Blockchain
Proof of PUF-Enabled Authentication (PoP) [17]	Manual	192.3	Full	Yes	Predefined PUF keys verification	Private	Blockchain
Proof of Block and Trade (PoBT) [6]	Manual	80-210	Full	Yes	Smart Contract and BFT	Private	Blockchain
McPoRA (Previous Paper) [1]	Manual	3.9-19.23 (Avg.)	Portion	No	UID verification	Private	Multichain
Minerless FlexiChain (Current Paper)	Automated 0.48 - 0.7 (Avg.)	1.23 - 3 (Avg.)	Portion	No	UID verification	Private	FlexiChain (Multiple-Integrated Conventional Blockchains)

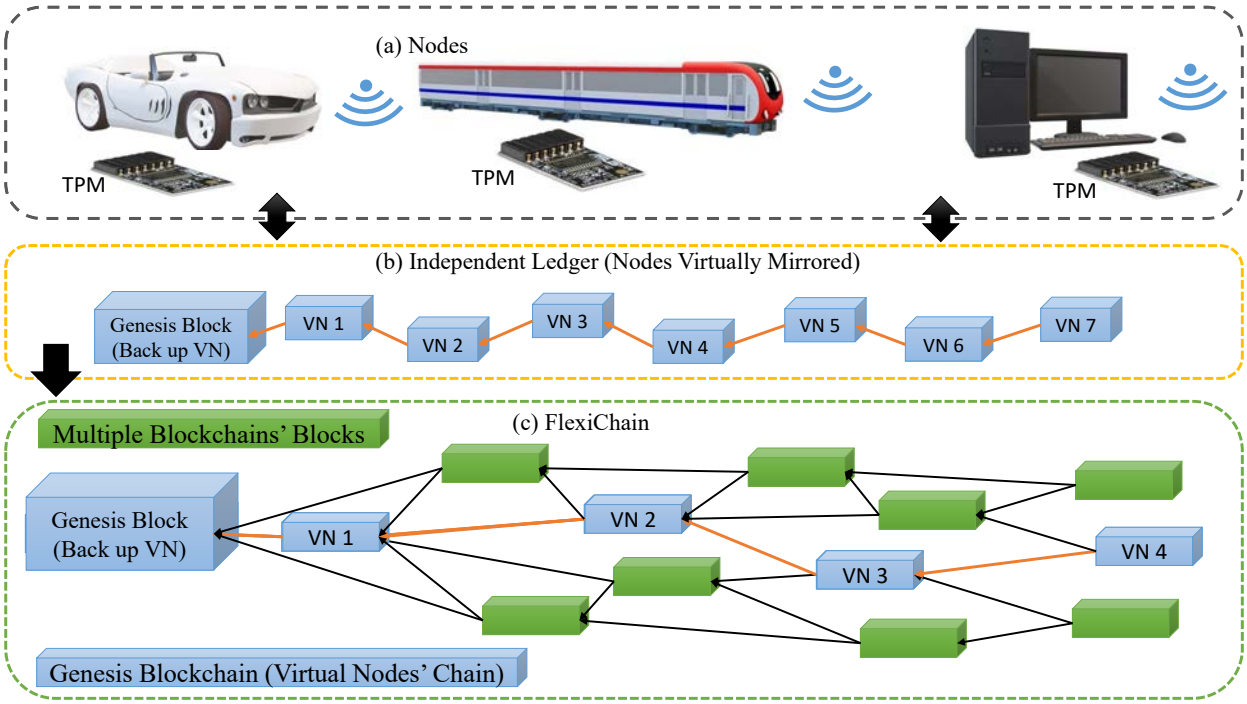


Fig. 4 Framework Illustration

(UIDs) which are used for the authentication operation within FlexiChain during the exchange of digital assets. The constructed UIDs are a series of linked UIDs where each UID is linked to the previous node's UID thus creating a chain of UIDs within the ledger. Figure 4 represents the framework of the current work and illustrates the proposed FlexiChain. The first level of the figure shows the nodes required to be represented in FlexiChain. The second layer represents the mirrored version of the represented nodes. The third layer illustrates the growing network and how the VN blockchain is integrated.

4.1 Consensus Algorithm

This part of the technology performs the registration and authentication processes and how they reach a consensus over a joining node or exchanged assets. Figure 6 represents types of blocks used in the proposed work.

For the registration process, block type A will be used and Algorithm 1 will recognize the block type and complete the process in Algorithm 2. During the operation, the system needs the Unique Identification Generator (UIDG) to create a new UID. By this step, the algorithm will refer to algorithm 3.

The type of transaction will be the indicator for the authenticator which type of blocks will be used and what process should be followed. Once the block type is recognized, it will be able to determine the keys and the algorithm to sign and append the block to two previous blocks by authenticating them similarly by their type. The confirmation of the authentication will be a signature (UID) of the authenticator node. Once the chain of narration of the block reaches the total number of active nodes, the block will be reduced to the minimal version. In section 4.1.1 the registration process will be explained in detail and in section 4.1.2 the authentication process will be clarified.

4.1.1 Registration

The registration process is presented in figure 5 for clarity. Node C wants to join the network and will use its own security hardware keys to register. The trusted modules' public keys ($TMpks$) are all predefined to the participants by their trusted modules. Two A type transactions will be broadcast to the whole network: $SigATrx1 := sign(sk, ATrx1(NewConstructedpID))$ and $SigATrx2 := (sign(sk, ATrx2(Parameters)))$ (Algorithm 1). Once the system recognizes the type of blocks and the keys used, the system will shift the rest of the process to Algorithm 2, as follows. The Genesis Node D will pick the transactions based on time and type, as presented in Algorithm 2. Node D authenticates $ifNodeCpk == TMpks(n)$, updates pID and generates a UID. Algorithm 3 presents the UID generation process. Node C updates the ledger and the UID should be part of the new virtual node header. The block should contain two previous hashes, one of which has the same type of blocks (figure 6). The design goal of this registration process is to ensure the integrity and security of the nodes by creating a UID for each one and constructing a new pair of IDs to use in the authentication process for other blocks, as elaborated in Algorithm 1. Since all node IDs are linked by a Merkle tree embedded in the blocks, any changes in the virtual blocks' linked list will be obvious and detected. Moreover, if one node was able to authenticate the block and add it to the network, then all nodes authenticated the block. Thus, there is no need for more authentications. However, the growing ledger uses confirmations for reduction process, which is the process where the block reaches its minimal size by ensuring that the number of confirmations equals the number of participants.

4.1.2 Authentication

Node C participates in the network and send transactions type B $\hookrightarrow BTrxs$ (figure 6). Similarly, they will be gathered based on sender, type, time consensus and type of block $BTrxs(n) \hookrightarrow Bb$. In order to add the Bb block, it will follow algorithm 1 and will be appended to the network. The new block header should be comprised of the authenticator UID, and two other hashes. Every time the block receives confirmations, the authenticators' UID $AUIDS$ will be listed until $AUIDS = AllUIDs$. The block will be reduced if in the chain of narrations the number of confirmations a block receives $AllUIDs == AUIDS$.

4.2 Ledger

The ledger in FlexiChain consists of accumulated multiple transactions grouped in a distinct block organized based on time consensus in a topological order. FlexiChain is the compound ledger that comprises both the independent ledger and the transaction ledger in one stronger ledger.

4.2.1 Independent Ledger

The independent ledger represents the installation and registration process of the nodes, which are mirrored virtually in the FlexiChain by a certain trusted module's public keys and block type. The ledger starts contemporary with the installation step with a certain number of nodes. One of the arcs of any new joining nodes should be linked to this ledger and the other will be attached randomly.

4.2.2 Transactions' Ledger

This is the ledger that carries the exchanged and shared digital assets through the FlexiChain network. New blocks authenticate previous ones in order to be listed and to be authenticated. Using virtual nodes linked within the network, the authentication is needed only once, which will equal the whole number of nodes.

4.3 Trusted Modules

The trusted modules are used in this technology for registration. Each module has its predefined keys and one can recognize the other modules. They should be manufactured by a trusted source even though they are

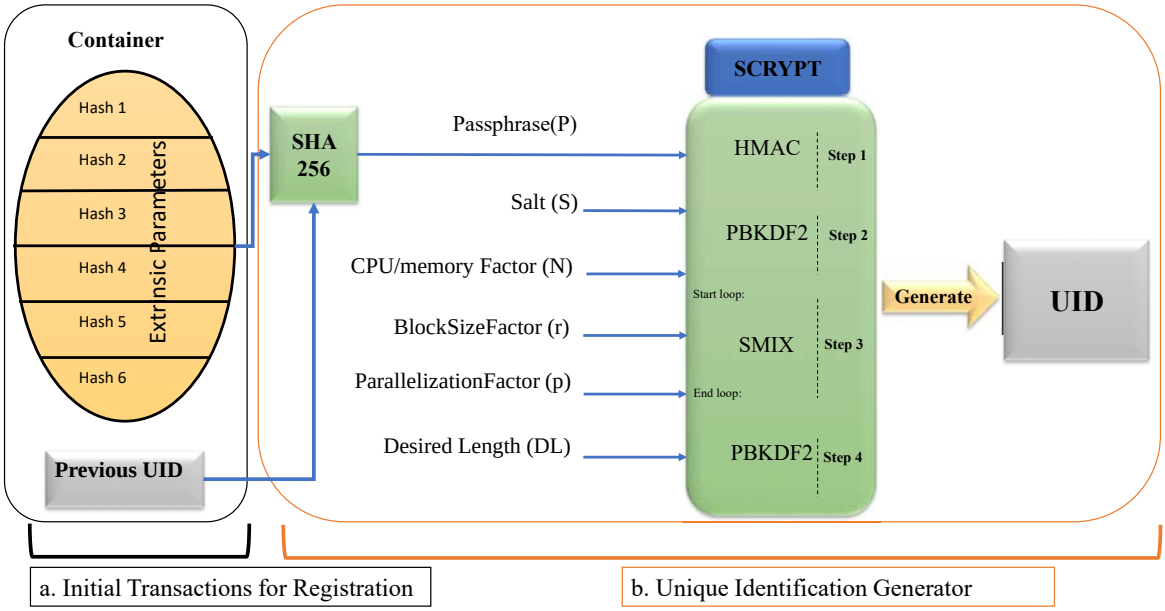
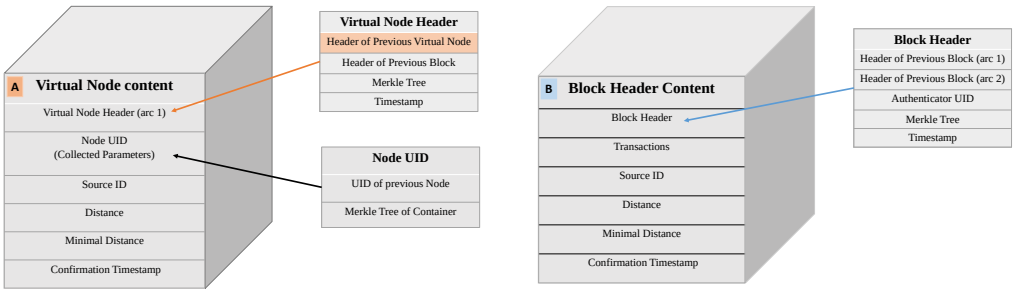


Fig. 5 Registration Process Depiction



(a) Block Type A (b) Block Type B

Fig. 6 Block Types in The Proposed Framework

just used one time through the whole process until each node can construct new IDs, which eventually will be used as major keys.

4.4 Block Type

The block type in this technology is an important factor and can be defined by the digital signatures used for this certain block or by its label. The blocks that have the same type are always linked randomly by one of its arcs. A certain block type represents a certain use purpose in the ledger. For this paper we are using two types of blocks: one which represents the virtual existence of nodes and preserves its UID, and the other type which is what the nodes are exchanging within the network.

4.5 Algorithms and Operations

Algorithm 1 of this technology will be executed based on the transaction,s label, which will be indicating whether the process to be taken is for a registration or trade. The first Algorithm handles the most operations within the network and is appointed to deal with type B blocks [1]. The initial few lines determine whether the appending steps for a certain block will be through algorithm 1 or algorithm 2.

Algorithm 2 presents the registration steps that use extrinsic parameters as an input to generate the UID through algorithm 3 and shift back to select a certain location that ensures one of the new blocks arcs should be connected to the genesis blockchain.

Algorithm 1 Proposed FlexiChain

Input : *Data D_i collected from node N_i*
Output : *Authenticated Blocks b_i or Discarded Blocks d_i*
Terms : *bc_n is blocks' number of authentication, n is the number of nodes*
`/* Node collects data, based on the transactions label A, or B */`
$N_i \hookrightarrow b_i$ Node N_i creates block b_iA or b_iB
if $b_i = A$, **then**
  Algorithm 2 `/* Shift to Algorithm 2 */`
  ,
else
  continue
end
Node runs Blocks Filtration Algorithm (BFA)
if $bc_i \equiv 0$ *in ledger,* **then**
  Pick b_{i_1} and b_{i_2} with $bc_i = 0$,
else
  Pick b_{i_1} and b_{i_2} with $bc_i = 0$ and $bc_i = 1$,
end
Pick b_{i_1} and b_{i_2} randomly
Node identifies two previous blocks as a location (l_i) $l_i \hookrightarrow b_i$ `/* Node checks the authenticity of the previous two blocks by comparing the predefined constructed UID derived from the ledger */`
if *UID in* $\overline{b_{i_2}}$ *and* $\overline{b_{i_1}} \neq$ *UIDs in SUIL* **then**
  Discard
else
  Authenticate
end
`/* Node broadcasts the new block to the network */`
N_i broadcasts block b_i `/* New block appended to DBL as a side block */`
$b_i \hookrightarrow DBL$
elf bc_i for each b_i in $DBL \equiv n$ Reduce Leave

Algorithm 2 Virtual Node (VN) Registration

Input : *Hash of Extrinsic Parameters and Constructed Public ID (Pd)*
Output : *Registered Virtual Node (VN)*

New Regular node (RN) joins the network RN constructs new pair IDs RN extracts extrinsic parameters RN broadcasts new constructed public ID Pd (Tx1) RN broadcasts a container of extrinsic parameters (Tx2) `/* All transactions are signed by trusted module keys (TMK) */`
BN receives Txs BN authenticates Txs **if** *public key = predefined public key* **then**
  BN claims Txs BN generates UID BN create a VN
else
  Discard Txs
end
BN generates UID BN hash container header and previous UID BN creates VN
- assign a timestamp
- assign UID `/* shift to Algorithm 2 */`
- assign public ID
- assign source ID

BN run BFA `/* locate the new block by choosing two previous blocks, one of them must be a VN */`
BN broadcasts VN BN updates ledger

Algorithm 3 UIDG Generation Algorithm [20]

Input : *P:String of characters in bytes*
Salt:Random salt in bytes
CostFactor(N):Integer CPU/memory cost
BlockSizeFactor(r):Integer blocksize
ParallelizationFactor(p):Parallelization
UID:Desired key length in bytes
Output : *UID:DesiredKeyLen long in bytes*

BN claims Txs `/* BN hash Tx1(Container) and previous UID */`
P = SHA256 (Container, Previous UID) `/* Setting the size of the Block */`
$128 * r$ *= Block Size Initial generation of Salt S used in PBKDF2* `/* Initial generation of random data r by PBKDF2 */`
*PBKDF2(P, S, 1, r*p)=[B_0.....B_{p-1}]* `/* The results will be used in the mixing function */`
for *i = 0 to p - 1* **do**
  MF(B_i,N)=B_i
end
`/* The output of the mixing function is the new expensiveSalt */`
ExpensiveS=[$B_0 \parallel B_1$..... $\parallel B_n$] `/* Generating UID which is the Result of the UIDG function */`
UID=PBKDF2(P, ExpensiveS, 1,UIDG)=[B_0.....$B_p - 1$] `/* shift to Algorithm 1 */`

5 Experimental Results

In this section, the simulation setup of the proposed technology and the results will be presented, analyzed and discussed. Python and PostgreSQL are the two major tools used in this experiment to create the virtual nodes and store transactions' hashes in the organized FlexiChain.

5.1 Simulation Setup

A P2P connection has been created between 10, 20, or 30 nodes. First, the registration process (see 4.1.1) will start by creating the independent ledger of at least two virtual nodes. The registered units will start to generate digital assets in a certain amount of time. Moreover, every few seconds a new node will join the network by performing the registration process concurrently with the exchange of the registered nodes. The nodes generating blocks will have to authenticate two previous blocks in order to add an unconfirmed block. PostgresSQL is used to store the block hashes and timestamps acquired from the process.

5.2 Time Analysis

5.2.1 Registration

Registration time is the time needed to enroll a node and give the node credentials to participate. Figure 7 represents the registration time per node for all scenarios. It is clear that there is a direct relationship between the number of nodes and total registration time. Each node consumed an average of 0.48, 0.54, or 0.7 ms/node. The registration average time reflects a fast and efficient process since it starts at millisecond levels.

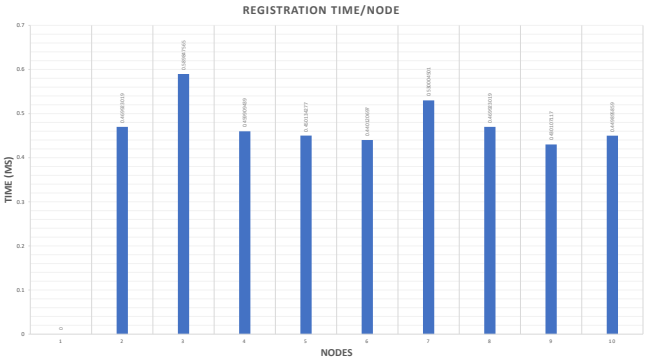
5.2.2 Authentication

Authentication time is the round time in ms needed to authenticate a block. In figure 8 a depiction of the average authentication time per transaction for all scenarios is shown. An inverse relationship between the number of nodes and authentication time is observed with an average of 3, 2.42, 1.23 ms/tx respectively. The average authentication time is suitable to CPS since it is close to real-time operations and runs in milliseconds. Moreover, the average time is decreasing over time which will speed up the reduction process and will eventually stabilize the registration process average time. The authentication time is presented in figure 8. The chart indicates that the growth of the nodes will increase the operations; thus, the average authentication time will decrease. The authentication time is a major factor of the CPS requirements listed in figure 1. Comparing the proposed paper to previous related works in 3, it is obvious that the targeted field of application is covered. General use protocols could be efficient but not applicable for all applications. Also, minerless FlexiChain presents very fast operations and a growing speed which will be more suitable for its targeted design.

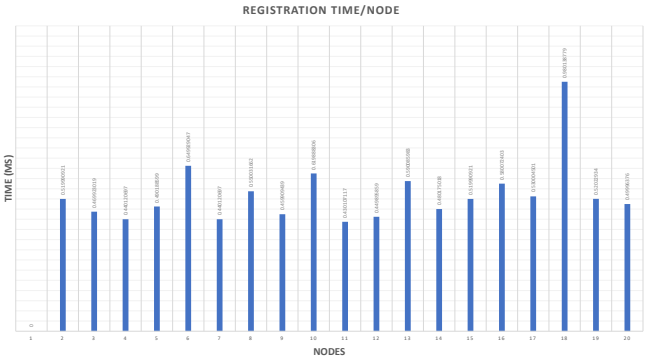
5.3 Security Analysis

In FlexiChain, both hardware and software security are taken into account by design to suit CPS applications. In this technology, various security issues could be tested to evaluate the technology. For example, an unregistered malicious node tries to join the network, a registered node that became a malicious node, physical security issues such as a hardware replacement, impersonation issue, and rainbow and brute force attacks.

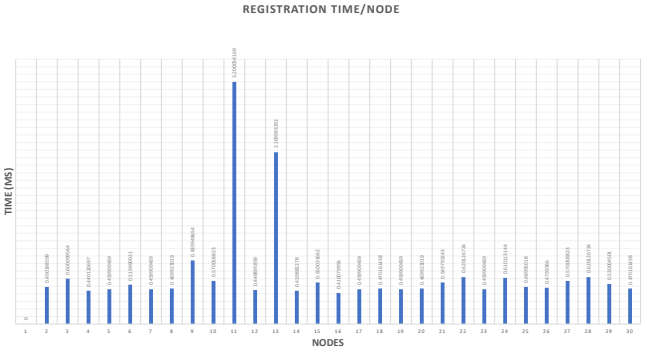
– **Unregistered Malicious Node**: If an unregistered node wanted to join the network, it will go through the registration protocol, which assumes the preexistence of the trusted module public key within the Back-up



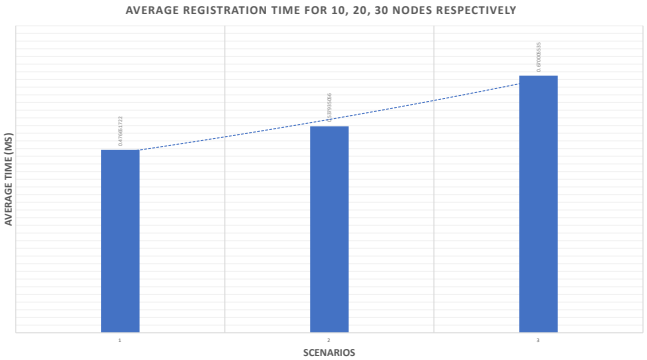
(a) Registration Time for Each Node (10 Nodes)



(b) Registration Time for Each Node (20 Nodes)

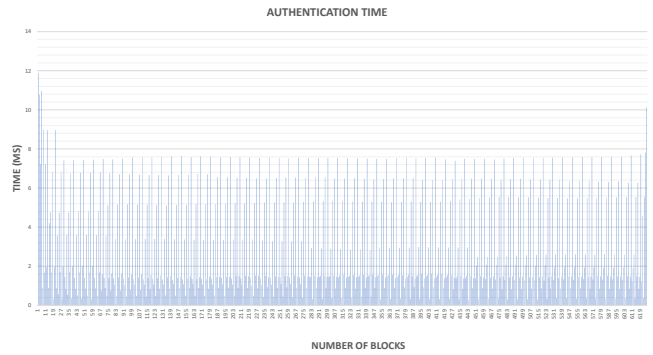


(c) Registration Time for Each Node (30 Nodes)

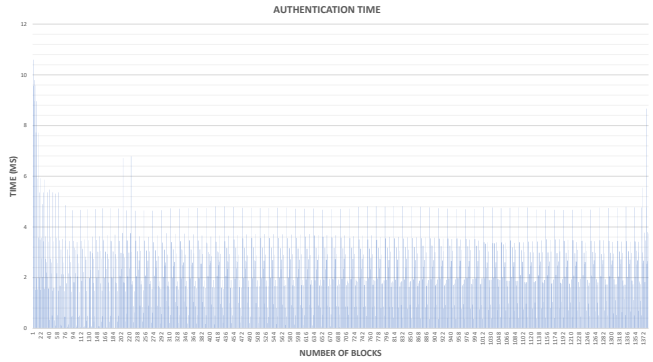


(d) Average Registration Time for All Scenarios

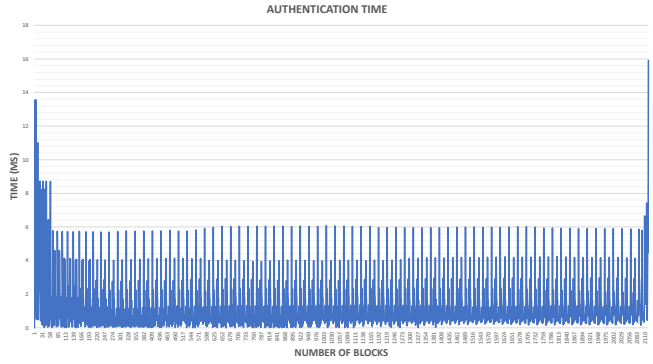
Fig. 7 Nodes Registration Time



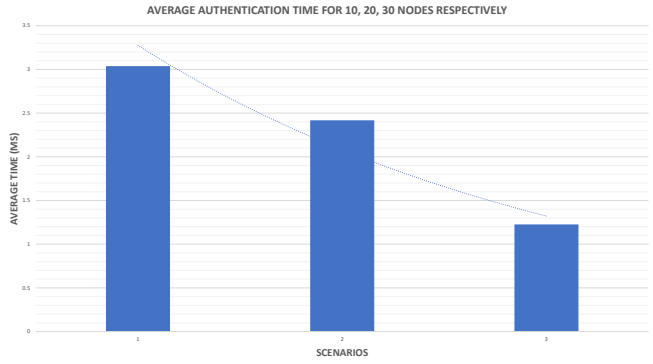
(a) Authentication Time (10 Nodes)



(b) Authentication Time (20 Nodes)



(c) Authentication Time (30 Nodes)



(d) Average Authentication Time for All Scenarios

Fig. 8 Authentication Time

Node (BN) list. Reaching the step where the BN should recognize the trusted module public key, the BN will compare the public key of the malicious node to the predefined list, and once the BN recognizes that this node does not have any predefined information to be updated, the BN will discard the request. Thus, the malicious node can not join the network.

– **Registered Malicious Node**: If a registered device has became a malicious node in the network, the UID of the same device will change based on the changes that took place in the same device, such as cloning the MAC address, updating the OS, or the firmware. This will change all the UIDs in the current node file that derived from the NodeChain which will avoid the node from authenticating other nodes' UIDs or even from getting its own blocks authenticated within the network.

– **Physical Attack**: Replacing a node or part of a node within the network will be detected through the extrinsic parameters that are extracted from each node in the registration protocol and used to generate the UID. This replacement will be reflected on the virtual existence of the same device. Any changes will result in deactivating the device and discarding any transactions.

– **Impersonation Attack** In case of impersonating a joined node, the attacker needs to impersonate all the specified independent ledger since the devices' UIDs are all connected to each other. Thus the process will be very expensive to impersonate one node.

6 Conclusions

DLTs have been widely used in various fields. Integrating DLT in CPS applications will increase security and integrity for safer applications and services. FlexiChain is designed in particular for IoT and CPS applications for security, scalability, speed and integrity. Two major parts are involved in the whole operation: one for registration, which is creating virtual versions of the actual nodes within the network, creating a linked list of UID, and using them in the authentication process which is the second part. In CPS applications, time is a major requirement since CPS requires nearly real-time operations. By using the linked list structure, and the consensus proposed, results are generated in milliseconds which indicates fast operations that could serve CPS environment.

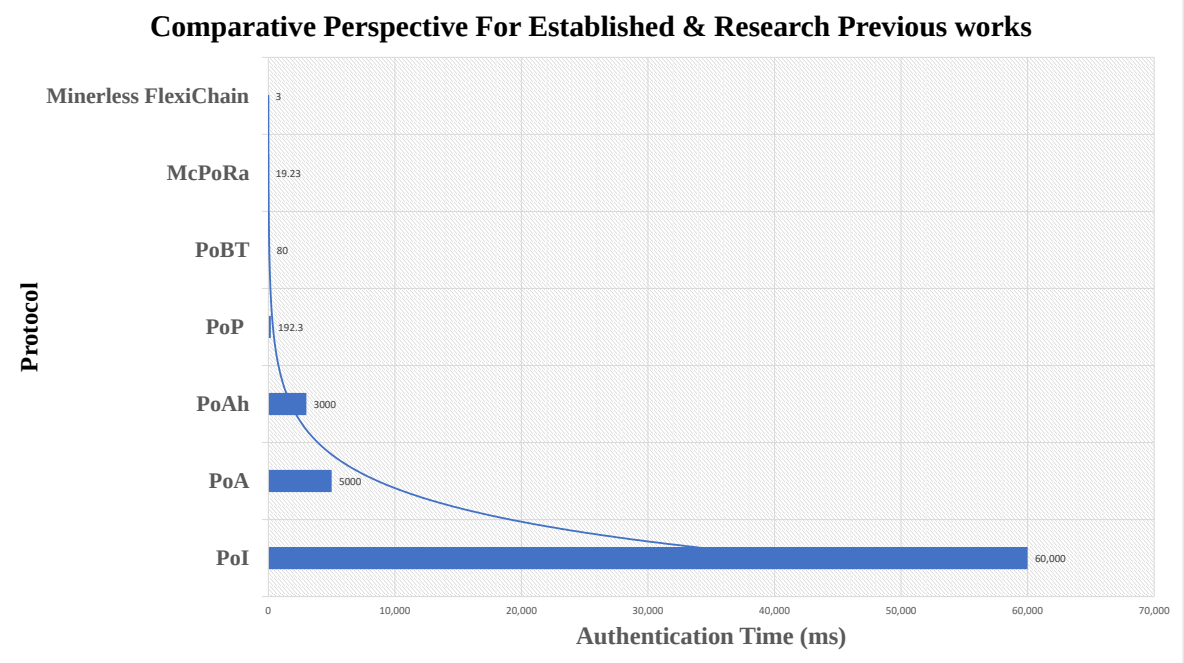


Fig. 9 Authentication Time Comparative Perspective Minerless FlexiChain vs Previous works

7 Future Directions

In our future work, Artificial Intelligence (AI) and smart contracts are targeted to enhance the overall performance. AI will be integrated in the selection algorithm to increase fairness and the growth of network. Smart contracts will be part of the registration process for robust access managed nodes mirroring ledger. Moreover, a Decentralized Autonomous Board (DAB) will be proposed and integrated with the initiation of the system. The future direction of the proposed paper is to enhance the protocol in terms of fairness, and embedded rules. Also, to propose the chain of narration mentioned briefly in this paper as an alternative to the registration process. Finally, expanding the number of participants and utilizing Docker tools for operations.

Compliance with Ethical Standards

The authors declare that they have no conflict of interest and there was no human or animal testing or participation involved in this research. All data were obtained from public domain sources.

References

1. Alkhodair, A., Mohanty, S., Kougianos, E., Puthal, D.: Mcpora: A multi-chain proof of rapid authentication for post-blockchain based security in large scale complex cyber-physical systems. In: 2020 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), pp. 446–451 (2020). DOI: 10.1109/ISVLSI49217.2020.00-16
2. Alkhodair, A.J., Mohanty, S.P., Kougianos, E.: Asid: Accessible secure unique identification file based device security in next generation blockchains. In: 2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), pp. 1–2 (2021). DOI: 10.1109/ICBC51069.2021.9461120
3. Baird, L.: The Swirlds Hashgraph Consensus Algorithm: Fair, Fast, Byzantine Fault Tolerance. Online (2016). URL `https://www.swirlds.com/`
4. Baird, L., Harmon, M., Madsen, P.: Hedera: A Public HashgraphNetwork & Governing Council (2019). URL `https://www.hedera.com/hh-whitepaper-v2.0-17Sep19.pdf`. Last Accssed on 21 Apr 2020
5. Biswas, S., Sharif, K., Li, F., Bairagi, A.K., Latif, Z., Mohanty, S.P.: Globechain: An interoperable blockchain for global sharing of healthcare data—a covid-19 perspective. IEEE Consumer Electronics Magazine **10**(5), 64–69 (2021). DOI: 10.1109/MCE.2021.3074688
6. Biswas, S., Sharif, K., Li, F., Maharjan, S., Mohanty, S.P., Wang, Y.: Pobt: A lightweight consensus algorithm for scalable iot business blockchain. IEEE Internet of Things Journal **7**(3), 2343–2355 (2020). DOI: 10.1109/JIOT.2019.2958077
7. Bodkhe, U., Mehta, D., Tanwar, S., Bhattacharya, P., Singh, P.K., Hong, W.: A survey on decentralized consensus mechanisms for cyber physical systems. IEEE Access **8**, 54371–54401 (2020). DOI: 10.1109/ACCESS.2020.2981415
8. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to algorithms, second edn. pp. 552–557. MIT Press and McGraw-Hill (2001)
9. Dedeoglu, V., Dorri, A., Jurdak, R., Michelin, R.A., Lunardi, R.C., Kanhere, S.S., Zorzo, A.F.: A journey in applying blockchain for cyberphysical systems. In: 2020 International Conference on COMmunication Systems NETworkS (COMSNETS), pp. 383–390 (2020). DOI: 10.1109/COMSNETS48256.2020.9027487
10. Egala, B.S., Pradhan, A.K., Badarla, V., Mohanty, S.P.: Fortified-chain: A blockchain-based framework for security

and privacy-assured internet of medical things with effective access control. IEEE Internet of Things Journal **8**(14), 11717–11731 (2021). DOI: 10.1109/JIOT.2021.3058946
11. Guin, U., Cui, P., Skjellum, A.: Ensuring proof-of-authenticity of iot edge devices using blockchain technology. 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), pp. 1042–1049 (2018). DOI: 10.1109/Cybermatics_2018.2018.00193
12. Živi, N., Kadušić, E., Kadušić, K.: Directed Acyclic Graph as Tangle: an IoT Alternative to Blockchains. In: Proc. 27th Telecommunications Forum (TELFOR), pp. 1–3 (2019)
13. Jungnickel, D.: Graphs networks and algorithms, fourth edn. pp. 92–93. Springer (2012)
14. King, S., Nadal, S.: PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. https://decred.org/research/king2012.pdf (2012)
15. LeMahieu, C.: Nano: A feeless distributed cryptocurrency network. White paper, Nano (2015). URL `https://content.nano.org/whitepaper/Nano_Whitepaper_en.pdf`
16. Mohanta, B.K., Satapathy, U., Dey, M.R., Panda, S.S., Jena, D.: Trust management in cyber physical system using blockchain. In: 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1–5 (2020). DOI: 10.1109/ICCCNT49239.2020.9225272
17. Mohanty, S.P., Yanambaka, V.P., Kougianos, E., Puthal, D.: Pufchain: A hardware-assisted blockchain for sustainable simultaneous device and data security in the internet of everything (ioe). IEEE Consumer Electronics Magazine **9**(2), 8–16 (2020). DOI: 10.1109/MCE.2019.2953758
18. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System. Online (2009). URL `https://bitcoin.org/bitcoin.pdf`
19. NemTeam: Nem technical reference. Tech. rep., NEM Blockchain (2018). URL `https://nemplatform.com/wp-content/uploads/2020/05/NEM_techRef.pdf`
20. Percival, C.: Stronger key derivation via sequential memory-hard functions. Online (2012). URL `https://www.tarsnap.com/scrypt.html`
21. Popov, S.: The tangle. Jinn Labs (2016). URL `https://www.iota.org/`. Version 0.6
22. Puthal, D., Malik, N., Mohanty, S.P., Kougianos, E., Das, G.: Everything you wanted to know about the blockchain: Its promise, components, processes, and problems. IEEE Consumer Electronics Magazine **7**(4), 6–14 (2018). DOI: 10.1109/MCE.2018.2816299
23. Puthal, D., Mohanty, S.P.: Proof of authentication: Iot-friendly blockchains. IEEE Potentials **38**(1), 26–29 (2019). DOI: 10.1109/MPOT.2018.2850541
24. Puthal, D., Mohanty, S.P., Nanda, P., Kougianos, E., Das, G.: Proof-of-Authentication for Scalable Blockchain in Resource-Constrained Distributed Systems. In: 2019 IEEE International Conference on Consumer Electronics (ICCE), pp. 1–5 (2019). DOI: 10.1109/ICCE.2019.8662009
25. Rahman, M.A., Rashid, M.M., Hossain, M.S., Hassanain, E., Alhamid, M.F., Guizani, M.: Blockchain and iot-based cognitive edge framework for sharing economy services in a smart city. IEEE Access **7**, 18611–18621 (2019). DOI: 10.1109/ACCESS.2019.2896065
26. Silvio Micali, M.R., Vadhan, S.: Verifiable random functions. Tech. rep., Foundations of Computer Science (1999). URL `https://www.cs.bu.edu/~goldbe/projects/vrf`
27. Skiena, S.S.: The algorithm design manual, second edn. pp. 495–497. Springer (2011)
28. Stanciu, A.: Blockchain based distributed control system for edge computing. In: 2017 21st International Conference on Control Systems and Computer Science (CSCS), pp. 667–671 (2017). DOI: 10.1109/CSCS.2017.102
29. Team: Proof-of-authority chains - wiki openethereum documentation. Tech. rep., OpenEthereum-Github (2021). URL `https://github.com/openethereum/openethereum`
30. Team, T.I.: Iotexa decentralized network for internet of thingspowered by a privacy-centric blockchain. White paper (2018). URL `https://iotex.io/research`
31. Thulasiraman, K., Swamy, M.N.S.: Graphs: theory and algorithms. p. 118. John Wiley and Son (1992)
32. Viriyasitavat, W., Xu, L.D., Bi, Z., Hoonsopon, D.: Blockchain technology for applications in internet of things—mapping from system design perspective. IEEE Internet of Things Journal **6**(5), 8155–8168 (2019). DOI: 10.1109/JIOT.2019.2925825