Designing Secure Lightweight Blockchain-Enabled RFID-Based Authentication Protocol for Supply Chains in 5G Mobile Edge Computing Environment

Journal:	*IEEE Transactions on Industrial Informatics*
Manuscript ID	TII-19-0621
Manuscript Type:	SS on Blockchain and AI enabled 5G Mobile Edge Computing
Keywords:	5G Mobile Edge Computing, authentication, blockchain, RFID, supplychain, security

Designing Secure Lightweight Blockchain-Enabled RFID-Based Authentication Protocol for Supply Chains in 5G Mobile Edge Computing Environment

***Abstract*—Secure and real-time data about goods in transit in supply chains needs bandwidth having capacity that is not fulfilled with the current infrastructure. Hence, 5G-enabled Internet of Things (IoT) in mobile edge computing is intended to substantially increase this capacity. To deal with this issue, we design a new efficient "lightweight blockchain-enabled RFID-based authentication protocol for supply chains in 5G mobile edge computing environment", called LBRAPS. LBRAPS is based on "bitwise exclusive-or (XOR)", "one-way cryptographic hash" and "bitwise rotation operations" only. LBRAPS is shown to be secure against various attacks. Moreover, the simulation-based formal security verification using the broadly-accepted "Automated Validation of Internet Security Protocols and Applications (AVISPA)" tool assures that LBRAPS is secure. Finally, it is shown that LBRAPS has better trade-off among its security and functionality features, communication and computation costs as compared to those for existing protocols.**

***Index Terms*—5G mobile edge computing, RFID, authentication, blockchain, supplychain, security.**

I. Introduction

5G is a combination of various technologies as well as mechanisms that is expected to land into the future networks to fulfill the uttermost capacity and performance demands. It is expected that the design of 5G networks would spin around "virtualization and programmability of networks and services" [1]. It is visualized that transition to 5G will be smoothed by today's emerging technologies, such as "Blockchain, Software Defined Networking (SDN), Network Functions Virtualization (NFV), Internet of Things (IoT), Mobile Edge Computing (MEC) and Fog Computing (FC)". In addition, SDN and NFV support new tools that will strengthen pliability in designing networks [2].

The use of blockchain technology enhanced with the power of 5G will serve not only to save companies millions of dollars in operating costs, but also in its potential legal fees arising from disputes that could have been avoided. For example, consider the typical supply-chain process. A smart contract prototype can streamline the supply-chain process and allow the automatic payment of goods upon receipt, and eliminate the need of having deal with accounts receivables, waiting a 30-day period for payment of goods received, and paying for billing department personnel to track down distributors with outstanding invoices [1]. With the use of blockchain technology coupled with the power of 5G, a shipment can be tracked so that both the manufacturer as well as the distributor instantly know exactly where they stand with respect to a volume incentive rebate.

A blockchain is treated as a "distributed database" that stores a chain of data packaged in sealed blocks in a secure & unchanging way serially. The block chain, called as a "ledger", is continually increasing. The new blocks are appended to the end of the block, and each new block mentions to the content of the "previous block". The blockchain users can either generate randomly or pre-define the block content. A set of transactions (data block) is cryptographically protected by the use of a "collision-resistant cryptographic one-way hash function", such as "Secure Hash Algorithm (SHA 256)" in order to ensure "anonymity, immutability and compactness of the block". In addition, the ledger along with its contents are reproduced and synchronized in a "Peer-to-Peer (P2P)" network across several peers, which will create a "distributed ledger". Even if the blockchain is a part of "Distributed Ledger Technologies (DLT)", a chain of blocks is not also utilized by all DLTs [3]. Such a technology is referred to as the blockchain.

Three types of blockchain are there: 1) "permissioned blockchains", 2) "permissionless (public) blockchains", and 3) "consortium blockchains". The trademarked networks involved in "permissioned blockchains" are the networks where the individuals (entities) can conduct transactions (e.g., a "group of banks processing financial transactions") [4].

The "distributed consensus" protocol assures that a majority of blockchain network peers agree on the precise condition of the shared ledger. Some of the used distributed consensus algorithms work without a central manager in the network, where the blockchain nodes accomplish the verification of a transaction in different manners of consensus, such as "PoW (Proof of Work), PoS (Proof of Stake), DPoS (Delegated Proof of Stake), PBFT (Practical Byzantine Fault Tolerance) and Raft" [4], [5], [6]. The major utilization of Blockchain is included in the Bitcoin and crypto-currency, where the PoW is utilized as a consensus protocol and the computing power as a system in order to determine the selected peer [4]. The basic architecture of the blockchain with distributed consensus is shown in Fig. 1 [5], [7].

In the following, we provide some key characteristics that are shared by the commercial transactions which use the blockchain technology.

1) Real-time records: Distributed ledgers are refreshed uninterruptedly as transactions and different occasions may happen with programming computerizing the procedure. Such features assure that each network entity should have its very own up-to-the-minute transaction records which will help to reduce the various opportunities for extortion.

2) Immutable records: Blockchain innovation empowers substances to create permanent and changeless transaction records. This capacity offers a conspicuous business advan-

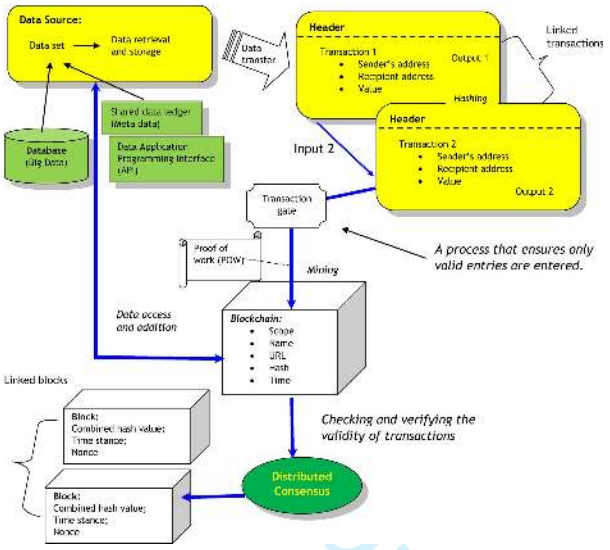

Fig. 1. Basic Architecture of Blockchain Technology [7]

tage, but it can likewise raise administrative hazard for some parties. Controllers can be offered consent to get to full transaction narratives in case of an inspection including transactions attached to a blockchain which make it increasingly troublesome for parties to contend that they lack sufficient transaction records.

3) Anonymity: Blockchain innovation creates it less demanding for the network users to be pseudonymous, which has repercussions for administrators of networks subject to various aspects, such as "anti-money laundering (AML)" and "know-your-customer (KYC)" regulations.

4) Cybersecurity risk: For an assortment of reasons, blockchain systems have turned out to be most loved focuses for hackers. The security incidents have ranged from ordinary administration disturbances to growing genuine burglaries of sensitive information as well as valuable crypto-currencies even if the decentralized blockchain structured networks make them stronger against various network-wide attacks including tampering of data.

5) Tax implications: Blockchain transactions including virtual currency can provide ascent to unforeseen tax consequences relying upon how the appropriate tax authority treats "virtual currency".

A. Motivation

The "Radio Frequency Identification (RFID)" technology has grown rapidly in recent years that has been adapted in various applications including "inventory management, supply chain, product tracking, transportation, logistics and self-administration store" [8]. Since the communication between a tag and a reader in RFID is via radio frequency signals (wireless) [9], an adversary can have several opportunity to perform passive attacks such as "eavesdropping attacks" and active attacks as well as "replay attacks and Denial-of-Service (DoS) attacks". To solve these issues, several common authentication security protocols have been suggested in the literature.

The existing protocols suggested in the literature need a database to be stored in the server side to help the authentication process. Apart from the multi-organization participation, we now consider another circumstance as the supply chain. Assume there are various departments or branches in a single organization, and specifically, some of them are geologically deployed or these are even appropriated in various nations. However, the various tasks and management related to an organization needs departments to allocate some information of the tags. In addition, a pre-requisite that new RFID framework as well as protocols should satisfy for an organization's practical requirements is to assure the privacy of the departments. Moreover, the synchronization is also unwieldy issue in distributed RFID frameworks when a new tag is incorporated or in each round the authentication message is refreshed. Consequently, this paper proposes a new blockchain-based mutual authentication RFID protocol that can fulfill the above needs [10].

RFID has several industrial applications today, such as "supply chain management, automated payment systems and airline baggage management". The sensitive RFID data is transmitted over the public Internet and also stored in various devices. Therefore, it is essential to have communication security protocols which make RFID systems because RFID enabled devices deal with various sensitive objects (e.g., passports and identity documents). Also, it is needed that the confidential RFID data should not be leaked in case of the real-time applications or health care monitoring system.

We set forward the necessities that the distributed *RFID* framework requirements are not yet accessible in the existing protocols. A multi-department collaboration situation is considered to exhibit a "blockchain-based distributed RFID framework" model and then to depict the proposed authentication protocol. Through the proposed blockchain-based scheme, the following objectives are accomplished at the same time: 1) security against several attacks; 2) "traceable and unmodifiable communication records"; 3) each department needs "its own secret tag information which is not included in the servers"; and 4) "inter-department sharing of insensitive tag information for authentication without central server or trusted third parties".

B. System Models

The requirements of RFID system within a company can be met in a "private blockchain with *Raft* consensus mechanism". Here, we consider the "multi-department cooperation distributed RFID" system that is illustrated in Fig. 2. Multiple departments keep a private blockchain together and also execute the same authentication procedure [11].

1) Authentication Model: As shown in Fig.2, we assume that there are N departments in the blockchain. Each department is privileged with nodes (Supplychain) and accounts (Reader-Tag). Here, the reader R initiates the process with a certain key for tag verification to prove its identity. The tag T proceeds further computations with the current timestamp and balance in the blockchain, and gives a challenge to the reader R. If the challenge is successful, the reader computes and

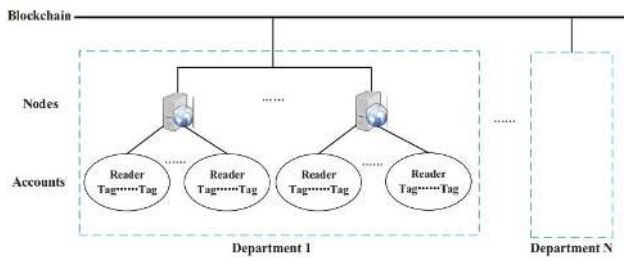

Fig. 2. Multi-department cooperation distributed RFID architecture [11]

processes the authentication message to the supply chain. The supplychain gets the message from the reader and validates the received message. Once it is proven to be successful, the supplychain acknowledges the supply amount and goes on to establish the session key which can be shared with the tag for future communications. This entire process happens via the reader R, and both the supplychain node and the tag T establish a session key and update the blockchain balance successfully.

2) Threat Model: Similar to any other networks, we apply the widely-accepted "Dolev-Yao (DY) threat model in which an adversary can not only can eavesdrop the communicated messages among the various entities, but can also modify, delete or insert fake messages in between the communication" [12] including to perform several potential attacks including "impersonation, replay, man-in-the-middle and Ephemeral Secret Leakage (ESL) attacks". The "Canetti and Krawczyk's adversary model (CK-adversary model)" [13] is a current *de facto* stronger model as compared to the DY model, which is also applied in many recent authentication protocols in the literature. The CK-adversary model allows "the adversary apart from his/her the abilities as performed using the DY model, he/she can also compromise the session states along with secret information including secret keys". Thus, even if the session states along with secret information are compromised in a particular session, these compromised information must not lead to compromise the secrecy of other parties' credentials. Hence, it is also important that under the CK-adversary model the forward & backward secrecy need to be preserved in an authentication protocol.

C. Research Contributions

We list the main research contributions as follows.

- A new lightweight blockchain enabled RFID-based authentication protocol, called LBRAPS, has been proposed, which is based on bitwise exclusive-or (XOR), one-way cryptographic hash and bitwise rotation operations. LBRAPS contains two phases, namely initialization and authentication & key agreement.
- LBRAPS is examined for its security part against various attacks against an active (passive) adversary.
- The simulation-based formal security verification using the broadly-accepted "Automated Validation of Internet Security Protocols and Applications (AVISPA)" tool assures that LBRAPS is also secure.
- LBRAPS has better trade-off among its security and functionality features, communication and computation costs as compared to those for other related existing protocols.

The paper is outlined as follows. The related work is discussed in Section II. The proposed blockchain-based authentication protocol (LBRAPS) is discussed in detail in Section III. The evaluation of LBRAPS for its both security is done in Section IV. The formal security verification using AVISPA-based software simulation tool is provided in Section V. In Section VI, we present the efficiency of LBRAPS in terms of computation and communication costs. Finally, the concluding remarks are highlighted in Section VII.

II. Related Work

The blockchain technology has several benefits for the supply chain application. However, there are many barriers to its widespread adoption. The "security and privacy issues associated with the integration of RFID technology in the blockchain system" is considered as one of the important barriers.

Toyoda et al. [14] designed a "blockchain-based product ownership management system of RFID-attached products for anti-counterfeits" in order to apply it in the post supply chain. They designed a full-fledged security protocol which can enable each entity, including supply chain partners as well as customers for transferring and proving the "ownership of RFID tag-attached products based on Electronic Product Code (EPC)". Since the EPC is transmitted as a fixed component during the entire process, an adversary can monitor the movement of the RFID tag-attached products based on the transmitted EPC value.

Mujahid *et al.* [15] proposed an "ultra-lightweight primitive, called as the pseudo-Kasami code". In their primitive, the secrecy for RFID systems is achieved by means of utilizing the unpredictable property of secret keys. Besides this, they proposed a mutual authentication RFID protocol based on their pseudo Kasami-code, bitwise XOR, bitwise rotation, and Hamming weight. Another RFID protocol, known as Gen2V2 proposed in [16], adds an extra security feature (called the untraceable command) to the protocol [15]. The "untraceable command" entitles a tag to reveal its secret credentials, such as EPC and user memory to restricted readers only. Since any unauthorized reader assembling with Gen2V2 protocol can demand as a privileged reader itself and also can undo a tag's untraceable feature, such security feature (untraceable command) leads to security attacks.

A "low-cost authentication protocol for the distributed database RFID system", called the HGLAP protocol, was also proposed in [17]. HGLAP is efficient because it helps in reducing the search time for a tag identity in the back-end database. The CRMAP protocol designed in [18] is a kind of "challenge-response authentication protocol", which relies on a "cryptographic collision-resistant hash function". CRMAP was shown to be robust against spoofing as well as replay attacks. Though there are some protocols proposed for distributed RFID systems [17], [18] in the literature, these

protocols hinge on either a "distributed database" or a "central trusted server".

Sidorov *et al.* [19] designed an "ultralight-weight RFID Protocol for blockchain enabled supply chains". However, the main issue related to their design was that an adversary can easily obtain the tactful credentials by means of capturing the communicated messages as it relies on only bitwise rotate operation. Masoumeh and Mahyar [20] also recommended that single or multiple applications of "bitwise rotate (ROT)" with "bitwise XOR" operations do no converge to build a secure protocol. Therefore, to construct a secure protocol without any cryptographic primitives is difficult. As a result, we feel that there is a requirement to construct a "robust and efficient RFID protocol" that can abolish the security flaws that are still found in the previous protocols for the purpose of integrating it with blockchain infrastructure.

III. The Proposed Scheme

In this section, we aim to construct a new lightweight blockchain enabled RFID-based authentication protocol, called LBRAPS. The LBRAPS is categorized into two phases: 1) initialization phase and 2) authentication & key establishment phase. The illustration of the LBRAPS is shown in Fig.3. Furthermore, Table I shows the notations and their significance, which are used in LBRAPS.

TABLE I
NOTATIONS/SYMBOL USED IN LBRAPS

Symbol	Description
R, T, S	Reader, tag and i^{th} supply chain node, respectively
ID_R, ID_T, ID_S	Identities of reader, tag, supplychain node, respectively
X_{RS}	A secret key between S and R
$Dept_i$	i^{th} department
Bal_{BC}	Balance amount in blockchain under $Dept_i$
B_s	Blockchain associated with S
$h(\cdot)$	"Collision-resistant" cryptographic one way hash function
SK	Session key between two entities
R_N, R_a, R_b, S_R	Random nonces
T_R, T_T, T_S	Current timestamps
ΔT	Maximum transmission delay
$ROT(X,Y)$	Left rotate of X by Hamming weight of Y
$RROT(X,Y)$	$X \ggg Y$, right rotate of X by Hamming weight of Y
$E_i \stackrel{?}{=} E_j$	Whether expression E_i equals to expression E_j
$\|, \oplus$	Concatenation and Bitwise XOR operations
$\mathcal{A}$	Adversary

A. Initialization Phase

To initialize the protocol, we consider the tag or reader ID as the password, and for each account identifier the blockchain generates public key address. Therefore, the tag stores the tuple $\{ID_T, Bal_{BC}\}$, where the tag ID and balance amount in blockchain under $Dept_i$ are ID_T and Bal_{BC}, respectively. Similarly, each reader also stores $\{ID_R\}$ in its memory, the reader ID is ID_R. Furthermore, the supply chain node (S) and the reader (R) will share a secret key $X_{RS} = h(ID_S \| B_s \| ID_R)$ which is private, where B_s represents the blockchain associated with S. Since reader (R) initiates the transaction and sends the transaction request to the tag (T), the reader R's account must have balance initially while creating account or mining which is realistic in nature. Therefore, the balance of each tag account in the blockchain is Bal_{BC} and for any new transactions it is initialized as $Bal_{New} = Bal_{BC} + S_{Amount}$, where S_{Amount} is the amount related to the supply chain transaction.

B. Authentication and Key Agreement Phase

The phase is executed by the participants: the reader (R), the tag (T) and the supply chain node (S), which is described by the following steps for mutual authentication purpose and also establishment of session key between T and S:

Step 1: $R \rightarrow T$: $MSG_1 = \{M_R, C_R, T_R\}$
The reader (R) generates a random number R_N and current timestamp T_R. Further, it computes $M_R = ROT(R_N \oplus ID_T \oplus T_R, T_R \oplus ID_T)$ and $C_R = h(M_R \| ID_T \| R_N)$, and then transmits the request message $MSG_1 = \{M_R, C_R, T_R\}$ to the tag (T) via open channel.

Step 2: $T \rightarrow R$: $MSG_2 = \{C_T, Auth_R, M_T, T_T\}$
The tag T receives the message MSG_1 from the reader (R) first checks the validity of timestamp T_R using the criteria $|T_R - T_R^*| < \Delta T$, where T_R^* and ΔT are receiving time of the message MSG_1 and "maximum allowable transmission delay", respectively. If it fails, the phase is instantly terminated by T. Otherwise, T extracts the random number R_N of R as $R'_N = (M_R \ggg (ID_T \oplus T_R)) \oplus ID_T \oplus T_R$, and calculates $C'_R = h(M_R \| ID_T \| R'_N)$ and checks $C'_R \stackrel{?}{=} C_R$. If it is valid, T also computes $C_T = h(R_N \oplus ID_T \oplus Bal_{New})$, $M_T = ROT(R_N \oplus ID_S \oplus T_T, T_T \oplus ID_T)$ and $Auth_R = h(C_T \| R_N \| M_T \| ID_T \| T_T)$. After this computations, the tag T sends the response message $MSG_2 = \{C_T, Auth_R, M_T, T_T\}$ to the reader R via open channel.

Step 3: $R \rightarrow S$: $MSG_3 = \{M_Q, M_P, Reader_{check}, T'_R\}$
The reader R receives the message MSG_2 from the tag (T) and verifies the authenticity of the received message by validating the timestamp T_R. If it holds, R further checks if $Auth_R \stackrel{?}{=} h(C_T \| R_N \| M_T \| ID_T \| T_T)$. If it also valid, R then generates two random nonces R_a and R_b at time T'_R, and computes $M_P = R_a \oplus ID_S \oplus R_b$, $M_Q = X_{RS} \oplus R_b$, and $Reader_{check} = h(R_a \oplus ID_S \oplus Bal_{New} \oplus (R_b \| T'_R))$. After this computations, the reader R sends the message $MSG_3 = \{M_Q, M_P, Reader_{check}, T'_R\}$ to the supply chain node S belongs to the department $Dept_1$ of the blockchain.

Step 4: $S \rightarrow R$: $MSG_4 = \{S_P, S_Q, S_S, T_S\}$
The supply chain S receives the message MSG_3 from the reader R and checks the validity of timestamp T'_R. If it is valid, S automatically starts the pre-defined smart contract on the blockchain to continue the authentication process. The authentication process is initiated first through the supplychain of blockchain by checking if the ID_T is in S's database. If it is not there, the process is terminated. Otherwise, S gets Bal_{BC-REC} and undergoes the following computations: $R_b = X_{RS} \oplus M_Q$, $R_a = M_P \oplus ID_S \oplus R_b$, $S_{check_A} = h(R_a \oplus ID_S \oplus Bal_{BC-REC} \oplus (R_b \| T'_R))$ and $S_{check_B} = h(R_a \oplus ID_S \oplus (Bal_{BC-REC} + S_{Amount}) \oplus (R_b \| T'_R))$. Once these computations are performed, the verification check is done by checking the condition $(S_{check_A} = Reader_{check})$

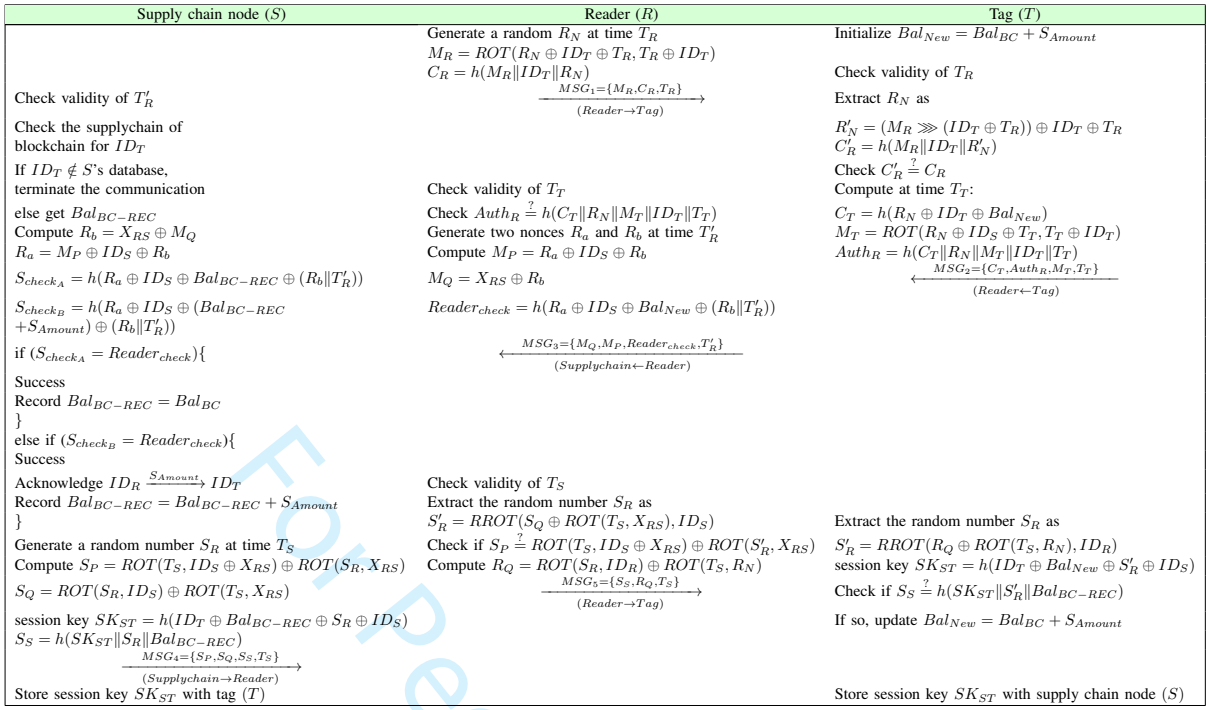

Fig. 3. Summary of mutual authentication and key agreement phase in LBRAPS

and if it successful, S records $Bal_{BC-REC} = Bal_{BC}$. Otherwise, if ($S_{check_B} = Reader_{check}$) is valid, S acknowledges $ID_R \xrightarrow{S_{Amount}} ID_T$, and also records $Bal_{BC-REC} = Bal_{BC-REC} + S_{Amount}$ in the distributed ledger $Ledger_{BC}$. Furthermore, S generates a random number S_R at current timestamp T_S to compute $S_P = ROT(T_S, ID_S \oplus X_{RS}) \oplus ROT(S_R, X_{RS})$, $S_Q = ROT(S_R, ID_S) \oplus ROT(T_S, X_{RS})$, $SK_{ST} = h(ID_T \oplus Bal_{BC-REC} \oplus S_R \oplus ID_S)$ and $S_S = h(SK_{ST} \| S_R \| Bal_{BC-REC})$. Here, SK_{ST} is the session key initiated by the supply chain node S so that the tag (T) can also establish the same session key by authenticating the valid messages. The supply chain node S then sends the message $MSG_4 = \{S_P, S_Q, S_S, T_S\}$ to the reader (R) via open channel.

Step 5: $R \rightarrow T$: $MSG_5 = \{S_S, R_Q, T_S\}$
The reader (R) receives the message MSG_4 from the supply-chain node (S) and checks the validity of received timestamp T_S. If it does not fail, R extracts the random number S_R of the reader as $S'_R = RROT(S_Q \oplus ROT(T_S, X_{RS}), ID_S)$ and validates it to authenticate the supply chain node S by checking if $S_P \stackrel{?}{=} ROT(T_S, ID_S \oplus X_{RS}) \oplus ROT(S'_R, X_{RS})$. If it passes, the reader (R) further computes $R_Q = ROT(S_R, ID_R) \oplus ROT(T_S, R_N)$ and transmits the message $MSG_5 = \{S_S, R_Q, T_S\}$ to the tag (T) via open channel.

Step 6: The tag (T) receives the message MSG_5 from the reader (R) extracts the random number S_R of the supply chain as $S'_R = RROT(R_Q \oplus ROT(T_S, R_N), ID_R)$, computes session key $SK_{ST} = h(ID_T \oplus Bal_{New} \oplus S'_R \oplus ID_S)$ to authenticate both supply chain S and reader R by verifying the condition $S_S \stackrel{?}{=} h(SK_{ST} \| S'_R \| Bal_{BC-REC})$. If the verification fails, the tag T refuses the communication. Otherwise, on successful authentication, the tag (T) updates $Bal_{New} = Bal_{BC} + S_{Amount}$ in its record as well as in its database too.

After establishing the session key ($SK_{ST} = h(ID_T \oplus Bal_{New} \oplus S'_R \oplus ID_S)$) between the T and S with the help of the R, the blockchain balance is updated in the distributed ledger with new balance Bal_{New}. The motivation for establishing the session key between the tag T and the supply chain node S is that depending upon the future requirement the blockchain can intercept with the concerned department where the T and S want to communicate securely with the help of the established session key SK_{ST}. It is worth noting that as illustrated in Sidorov *et al.*'s protocol [19], the practical scenarios of our mutual authentication protocol (LBRAPS) in blockchain-enabled supply chain can be also made easily in a similar way for various departments.

IV. Security Analysis

This section shows that LBRAPS is resilient against several attacks, and also ensures user anonymity and untraceability.

1) Confidentiality: The messages $MSG_1 = \{M_R, C_R, T_R\}$, $MSG_2 = \{C_T, Auth_R, M_T, T_T\}$, $MSG_3 = \{M_Q, M_P, Reader_{check}, T'_R\}$, $MSG_4 = \{S_P, S_Q, S_S, T_S\}$, $MSG_5 = \{S_S, R_Q, T_S\}$ are related to the random numbers (R_N, R_a, R_b and S_R) generated by the participants. It is difficult for an attacker $\mathcal{A}$ to extract R_N, R_a, R_b and S_R to compute the correct messages and also to make believe the other participants as authentic. Therefore, $\mathcal{A}$ cannot acquire the random numbers from the messages and impersonate the legitimate entities. Hence, the confidentiality of the transmitted data is preserved in LBRAPS.

2) User Anonymity and Untraceability: According to the threat model discussed in Section I-B2, $\mathcal{A}$ can capture the messages MSG_j, $j = 1, 2, 3, 4, 5$, which were communicated during the authentication & key agreement phase over the

insecure channels. Without knowing the parametric values ID_R, ID_T and R_N, it is computationally infeasible task for $\mathcal{A}$ to guess the identities of both tag T and reader R in polynomial time. So, this ensures that $LBRAPS$ holds the user anonymity property. In ensuring untraceability property, it is worth noticing that the messages MSG_j $(i = 1, 2, 3, 4, 5)$ are "dynamic" in nature which were computed using random numbers and current timestamps. Furthermore, due to the "non-invertible (collision-resistant) one-way property" of hash function $h(\cdot)$, it is computational task for $\mathcal{A}$ to trace the messages. Therefore, $\mathcal{A}$ can not keep track of the activities performed by the reader R and tag T over different sessions. Hence, LBRAPS also assures untraceability property.

3) Mutual Authentication and Session Key Establishment: The reader R, the tag T, and the supply chain node S authenticate each other in LBRAPS. The message $MSG_1 = \{M_R, C_R, T_R\}$ is protected by the secret random number R_N and also the current time stamp T_R. Only the legitimate T can acknowledge by verifying if $C'_R \stackrel{?}{=} h(M_R \| ID_T \| R'_N)$. In the next step, T raises the challenge to the reader to authenticate the message $MSG_2 = \{C_T, Auth_R, M_T, T_T\}$ sent by T. The R then validates the message sent by the T whether $Auth_R \stackrel{?}{=} h(C_T \| R_N \| M_T \| ID_T \| T_T)$. Now, R sends the message $MSG_3 = \{M_Q, M_P, Reader_{check}, T'_R\}$ to the S. On receiving the message, the S generates its random number S_R and the current timestamp T_S, and computes the session key $SK_{ST} = h(ID_T \oplus Bal_{New} \oplus S'_R \oplus ID_S)$ to transmit the message $MSG_4 = \{S_P, S_Q, S_S, T_S\}$. Upon receiving the message from S, the R verifies the authenticity of S by validating the message with $S_P \stackrel{?}{=} ROT(T_S, ID_S \oplus X_{RS}) \oplus ROT(S'_R, X_{RS})$. On successful verification, R authenticates S and transmits the message $MSG_5 = \{S_S, R_Q, T_S\}$ to T. T extracts the random number S_R of R and verifies the authenticity of the message MSG_5 sent by R as $S_S \stackrel{?}{=} h(SK_{ST} \| S'_R \| Bal_{BC-REC})$. If the verification is successful, both T and S are successful in establishing the session key SK_{ST} and furthermore, all the participants are successful in verifying the authenticity of the sender and receiver parties. Thus, LBRAPS is successfully preserving mutual authentication and ensuring the establishment of the session key.

4) Forward Secrecy: If the tag T is compromised by an adversary $\mathcal{A}$, the secret keys stored inside it are also leaked. $\mathcal{A}$ may then acquire the information transmitted in previous session. In LBRAPS, the random numbers and shared secret keys are not stored on the tag's memory. Furthermore, the random numbers are generated freshly in every session for computing the session keys so that the session keys are distict in each session. Thus, compromising the session key or random numbers in a specific session do not lead to any advantage to $\mathcal{A}$. Hence, LBRAPS ensures the forward secrecy.

5) Internal Attacks: Internal member may cheat other entities by impersonating other legal participants. Therefore, in our LBRAPS, there are two types of internal attacks, which are described below.

i) *Reader Impersonation Attack:* We consider that an internal legitimate reader of department 1 $(Dept_1)$ with the secret parameters tries to impersonate the other reader of $Dept_2$ who owns the secret parameters that are different from $Dept_1$. Now, when the forged reader queries the tag, the forge reader may send MSG_1 to the legitimate tag which belongs to $Dept_2$. The message MSG_1 cannot be authenticated by the legitimate tag because the random numbers generated by the reader of $Dept_2$ are different from those generated by the reader of $Dept_1$. So, the tag cannot extract the correct random number R_N and also cannot verify the authenticity of the message MSG_1. Thus, the reader impersonation attack is restricted in LBRAPS.

ii) *Tag Impersonation Attack:* We also consider that an internal legitimate tag of $Dept_1$ with the secret parameters tries to impersonate the other tag of another $Dept_2$ who owns the secret parameters that are different from $Dept_1$. When the forged tag queries the reader, the forged tag sends MSG_2 to the legitimate reader which belongs to $Dept_2$. The message MSG_2 cannot be authenticated by the legitimate reader because the random numbers generated by the tag of $Dept_2$ are different from those generated by the tag of $Dept_1$. This means that the reader cannot verify the authenticity of MSG_2 by checking the condition $Auth_R \stackrel{?}{=} h(C_T \| R_N \| M_T \| ID_T \| T_T)$. Thus, the tag impersonation attack is also restricted in LBRAPS.

6) External Attacks: We consider the following two active attacks.

i) *Replay Attack:* We consider that during the authentication & key agreement phase, $\mathcal{A}$ tries to intercept the messages MSG_j, $j = 1, 2, 3, 4, 5$ to frame replay attack by replaying these messages to the receiver. But this attempt fails due to the involvement of the current timestamps and random numbers embedded in the communicated messages MSG_j, $j = 1, 2, 3, 4, 5$. Upon receiving the messages, the initial step is the timestamp verification and then for the validation of the transmitted messages. Thus, framing the replay attack is resisted in LBRAPS.

ii) *Man-in-the-Middle Attack:* During the authentication & key agreement phase, suppose $\mathcal{A}$ wishes to capture and modify the messages MSG_j, $j = 1, 2, 3, 4, 5$ to believe the participants that the messages received from the genuine authentic participants. To frame this attack, suppose $\mathcal{A}$ wants to modify the message MSG_1. But, this attempt fails due to the lack of knowledge on the involved secret R_N. Similarly, $\mathcal{A}$'s attempts also fail to modify the other messages: MSG_2 for random secret R_N; MSG_3 for random secrets R_a & R_b; MSG_4 for random secrets S_R; and MSG_5 for random secret S_R. Thus, due to randomness of the messages and usage of current timestamps, LBRAPS withstands the "man-in-the-middle attack".

7) Ephemeral Secret Leakage (ESL) Attack:: During the authentication & key agreement phase, after validating mutual authentication as shown above, both the supply chain node S and the tag T establish a common session key SK_{ST}. According to the discussed threat model in Section I-B2, we consider the current *de facto* CK-adversary model for the session key (SK)-security. The reliability of SK-security in LBRAPS is relied on the following two cases:

Case 1. Assume the ephemeral (short term) secrets R_N, R_a, R_b and S_R are some how known to an adversary $\mathcal{A}$. The challenge for $\mathcal{A}$ is to create the session key SK_{ST}

based on the short term secrets. But, due to the lack of knowledge of long term secrets (ID_T, ID_S, X_{RS}, ID_R, Bal_{BC} and S_Amount), $\mathcal{A}$ fails to succeed in its challenge as it is "computationally infeasible task" for $\mathcal{A}$ to guess the long term secrets.

Case 2. Suppose few or all of the long-term secrets (ID_T, ID_S, X_{RS}, ID_R, Bal_{BC} and S_Amount) are some how leaked to $\mathcal{A}$. Now, the similar challenge for $\mathcal{A}$ as in Case 1 remains to construct SK_{ST} based on the long term secrets. However, without knowledge of short term secrets (R_N, R_a, R_b and S_R), it is also "computationally infeasible task" for $\mathcal{A}$ to win the challenge by guessing only the short term secrets.

From the above two cases, it is clear that the valid session key SK_{ST} is only computed with legitimate long term secrets along with short term secrets, which is possible only by the legitimate participants (S, R, and T). Furthermore, in LBRAPS, compromising of current session key does not lead to compromise the previous and future sessions as the session keys are randoms and unique in each session. Therefore, $\mathcal{A}$ can not determine the previous and future session keys even if the current session key is compromised [21]. Thus, LBRAPS successfully preserves both backward and forward secrecy along with the SK-security. Even with the help of some session hijacking attacks, only a particular session key can be leaked. But, its effect does not compromise the previous & future sessions. Hence, LBRAPS is secure against ESL attack.

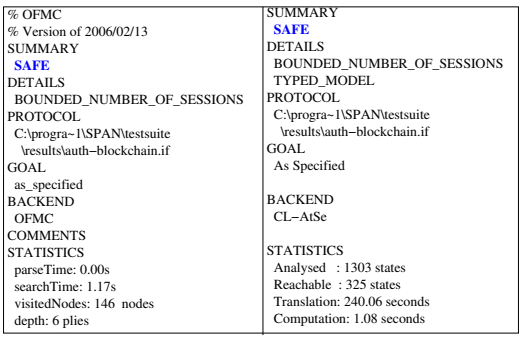

Fig. 4. Analysis of simulation results under OFMC & CL-AtSe backends

V. Formal Security Verification using AVISPA: Simulation Study

We apply the broadly accepted the software verification tool, called AVISPA [22] for validating the security of our proposed LBRAP against an adversary.

The "High-Level Protocol Specification Language (HLPSL)" in AVISPA is used to implement a security protocol in order to test whether the designed protocol is safe or unsafe using one of the four backends, namely "On-the-fly Model-Checker (OFMC), Constraint Logic based Attack Searcher (CL-AtSe), SAT-based Model-Checker (SATMC), and Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP)". The HLPSL code is transferred into the "Intermediate Format (IF)". The IF is then supplied as input to one of the four backends, which leads to produce the "Output Format (OF)". The OF has various sections as described in [22]. More details about AVISPA as well as HLPSL implementation can be found in [22].

In our implementation, we have basic and composite roles. The basic roles represent various participants in the protocol (the roles for the reader, tag and supply chain node). However, the composition roles, which are mandatory roles (session and goal & environment), are various scenarios involving basic roles.

The broadly accepted "SPAN (Security Protocol ANimator for AVISPA)" tool [23] is applied to perform formal security verification part through simulation on our LBRAP. The simulation results shown in Fig. 4 assure that LBRAP protects both replay & man-in-the-middle attacks.

VI. Performance Comparison

We perform a rigorous comparative study on "security & functionality features, computation and communication costs during the authentication & key agreement phase" among the proposed LBRAP and the existing schemes of Sidorov *et al.* [19] and Mujahid *et al.* [15].

1) Comparison of Security and Functionality Features: In Table II, LBRAPS is compared with the earlier schemes of Sidorov *et al.* [19] and Mujahid *et al.* [15] based on several "security & functionality features SFA_i $(i = 1, 2, \ldots, 11)$". It is evident that LBRAPS supports more functionality features and also provides better security features as compared to those for other schemes.

TABLE II
Comparison of Security & Functionality Features

Attribute	Sidorov *et al.* [19]	Mujahid *et al.* [15]	LBRAPS
SFA_1	×	×	√
SFA_2	√	×	√
SFA_3	√	√	√
SFA_4	√	×	√
SFA_5	√	√	√
SFA_6	×	×	√
SFA_7	√	√	√
SFA_8	×	×	√
SFA_9	×	×	√
SFA_{10}	√	×	√
SFA_{11}	√	×	√

√: a scheme supports an attribute or resists an attack; ×: a scheme does not support an attribute or it does not resist an attack.
SFA_1: privileged-insider attack; SFA_2: anonymity; SFA_3: traceability; SFA_4: denial-of-service attack; SFA_5: mutual authentication; SFA_6: ESL attack; SFA_7: replay attack; SFA_8: impersonation attacks; SFA_9: man-in-the-middle attack; SFA_{10}: formal security verification using AVISPA tool; SFA_{13}: whether blockchain enabled

2) Comparison of Communication Costs: We consider that hash output is 160 bits (if SHA-1 hash function [24] is applied), humming weight is 160 bits, identities and random numbers are 160 bits and timestamp is 32 bits. In the protocols of Sidorov *et al.* [19] & Mujahid *et al.* [15], the "Hello" message is considered as 160 bits. In our LBRAPS, the messages MSG_1, MSG_2, MSG_3, MSG_4 and MSG_5 need $(160+\ 160+\ 32) = 352$ bits, $(160+\ 160+\ 160+\ 32) = 512$ bits, $(160+\ 160+\ 160+\ 32) = 512$ bits, $(160+\ 160+\ 160+\ 32) = 512$ bits and $(160+\ 160+\ 32) = 352$ bits, respectively. Therefore, the total communication cost required in LBRAPS due to exchange of five messages is 2240 bits. We

then compare the communication cost of LBRAPS with other schemes in Table III. It is observed that the proposed LBRAPS needs more communication cost as compared to two other schemes. However, our LBRAPS is the only able to protect RFID systems from all potential security attacks. In addition, LBRAPS outperforms other existing protocols, and LBRAPS and Sidorov *et al.*'s scheme [19] are the protocols that were designed to be integrated into the blockchain whereas Mujahid *et al.*'s scheme [15] is not designed for blockchain. Moreover, in Mujahid *et al.*'s scheme no session key is established between the entities.

TABLE III
COMPARISON OF COMMUNICATION & COMPUTATION COSTS

Attribute	Sidorov *et al.* [19]	Mujahid *et al.* [15]	LBRAPS
Communication cost	1760 bits	960 bits	2240 bits
Exchanged messages	5	4	5
Computation cost	$15T_{HW} + 14T_{ROT}$ $+12T_{xor}$ ≈ 0.0048 s	$1T_{HW} + 29T_{ROT}$ $+29T_{xor}$ ≈ 0.00032 s	$12T_{Hash} + 15T_{ROT}$ $+25T_{xor}$ ≈ 0.00384 s

3) Comparison of Computation Costs: Let T_{xor}, T_{HW}, T_{Hash}, and T_{ROT} denote the time needed for executing an exclusive-OR, hamming weight, one-way hash function and left/right rotation operations, respectively. It is assumed that $T_{Hash} \approx T_{HW} \approx 0.00032$ seconds [21]. As T_{ROT} and T_{xor} are negligible in computation, these are ignored in computation. During the authentication & key agreement phase of LBRAPS, an RFID tag has the computational cost of $5T_{Hash} + 4T_{ROT}$, while a reader has a computational cost of $3T_{Hash} + 7T_{ROT}$ and the supply chain node has the computational cost of $4T_{Hash} + 4T_{ROT}$ for the supplychain of blockchain data update process. Therefore, the total computation cost of LBRAPS is $12T_{Hash} + 15T_{ROT}$. From the comparative study on computational costs among the schemes presented in Table III, LBRAPS needs less cost than Sidorov *et al.*'s scheme [19]. Though Mujahid *et al.*'s scheme [15] needs less cost than both LBRAPS and Sidorov *et al.*'s scheme, no session key is established between the entities in [15] and fails to support all functionality and security features (see Table II).

VII. CONCLUDING REMARKS

This article handles an important problem related to "blockchain-enabled RFID-based authentication for supply chains in 5G mobile edge computing environment". We proposed an efficient authentication protocol (LBRAPS), which is not only efficient in both communication & computation, but also supports many security and functionality features. Various potential attacks are protected in LBRAPS. The AVISPA-based simulation on the formal security analysis also proves that LBRAPS is secure against active attacks. Moreover, LBRAPS has better trade-off among "security and functionality features, communication and computational costs" as compared to other schemes which are demonstrated in Tables II and III.

REFERENCES

[1] J. N. Dewey, R. Hill, and R. Plasencia, "Blockchain and 5G-Enabled Internet of Things (IoT) will redefine Supply Chains and Trade Finance," https://www.hklaw.com/files/Uploads/Documents/Articles/Blockchain5GEnabledInternetofThings.pdf. Accessed on February 2019.

[2] S. Yi, C. Li, and Q. Li, "A survey of fog computing: concepts, applications and issues," in *Workshop on Mobile Big Data (Mobidata'15)*, Hangzhou, China, 2015, pp. 37–42.

[3] J. W. Michael, A. Cohn, and J. R. Butcher, "BlockChain technology," *The Journal*, 2018.

[4] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," in *IEEE International Congress on Big Data (BigData Congress)*, Honolulu, HI, USA, 2017, pp. 557–564.

[5] T. Aste, P. Tasca, and T. Di Matteo, "Blockchain Technologies: The Foreseeable Impact on Society and Industry," *IEEE Computer*, vol. 50, no. 9, pp. 18–28, 2017.

[6] D. Ongaro and J. Ousterhout, "In search of an understandable consensus algorithm," in *USENIX Annual Technical Conference (USENIX ATC'14)*, Philadelphia, PA, USA, 2014, pp. 305–319.

[7] H. Min, "Blockchain technology for enhancing supply chain resilience," *Business Horizons*, vol. 62, no. 1, pp. 35–45, 2019.

[8] P. Fraga-Lamas and T. M. Fernández-Caramés, "Reverse engineering the communications protocol of an RFID public transportation card," in *IEEE International Conference on RFID (RFID)*, Phoenix, AZ, USA, 2017, pp. 30–35.

[9] L. Pang, L. He, Q. Pei, and Y. Wang, "Secure and efficient mutual authentication protocol for RFID conforming to the EPC C-1 G-2 standard," in *IEEE Wireless Communications and Networking Conference (WCNC'13)*, Shanghai, China, 2013, pp. 1870–1875.

[10] C. Su, Y. Li, Y. Zhao, R. H. Deng, Y. Zhao, and J. Zhou, "A survey on privacy frameworks for RFID authentication," *IEICE Transactions on Information and Systems*, vol. 95, no. 1, pp. 2–11, 2012.

[11] S. Wang, S. Zhu, and Y. Zhang, "Blockchain-based Mutual Authentication Security Protocol for Distributed RFID Systems," in *IEEE Symposium on Computers and Communications (ISCC)*, Natal, Brazil, 2018, pp. 74–77.

[12] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, 1983.

[13] R. Canetti and H. Krawczyk, "Universally Composable Notions of Key Exchange and Secure Channels," in *International Conference on the Theory and Applications of Cryptographic Techniques– Advances in Cryptology (EUROCRYPT'02)*, Amsterdam, The Netherlands, 2002, pp. 337–351.

[14] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, "A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain," *IEEE Access*, vol. 5, pp. 17 465–17 477, 2017.

[15] U. Mujahid, M. Najam-ul Islam, and S. Sarwar, "A new ultralightweight RFID authentication protocol for passive low cost tags: KMAP," *Wireless Personal Communications*, vol. 94, no. 3, pp. 725–744, 2017.

[16] G. EPCglobal, "EPC radio-frequency identity protocols generation-2 UHF RFID; specification for RFID air interface protocol for communications at 860 MHz–960 MHz," *EPCglobal Inc., November*, 2013.

[17] J. Ha, H. Kim, J. Park, S. Moon, J. G. Nieto, and C. Boyd, "HGLAP–Hierarchical Group-Index Based Lightweight Authentication Protocol for Distributed RFID System," in *International Conference on Embedded and Ubiquitous Computing*. Taipei, Taiwan: Springer, 2007, pp. 557–567.

[18] K. Rhee, J. Kwak, S. Kim, and D. Won, "Challenge-response based RFID authentication protocol for distributed database environment," in *International Conference on Security in Pervasive Computing*. Springer, 2005, pp. 70–84.

[19] M. Sidorov, M. T. Ong, R. V. Sridharan, J. Nakamura, R. Ohmura, and J. H. Khor, "Ultralightweight Mutual Authentication RFID Protocol for Blockchain Enabled Supply Chains," *IEEE Access*, vol. 7, pp. 7273–7285, 2019.

[20] M. Safkhani and M. Shariat, "Implementation of secret disclosure attack against two IoT lightweight authentication protocols," *The Journal of Supercomputing*, vol. 74, no. 11, pp. 6220–6235, 2018.

[21] J. Srinivas, A. K. Das, N. Kumar, and J. Rodrigues, "Cloud centric authentication for wearable healthcare monitoring system," *IEEE Transactions on Dependable and Secure Computing*, 2018.

[22] AVISPA, "Automated Validation of Internet Security Protocols and Applications," 2019, http://www.avispa-project.org/. Accessed on February 2019.

[23] ——, "SPAN, the Security Protocol ANimator for AVISPA," 2019, http://www.avispa-project.org/. Accessed on February 2019.

[24] "Secure Hash Standard," FIPS PUB 180-1, National Institute of Standards and Technology (NIST), U.S. Department of Commerce, April 1995. http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf. Accessed on January 2019.

Cover Letter

Dear Editor-in-Chief and Guest Editors
IEEE Transactions on Industrial Informatics

Please accept this submission (**Designing Secure Lightweight Blockchain-Enabled RFID-Based Authentication Protocol for Supply Chains in 5G Mobile Edge Computing Environment**) to the IEEE Transactions on Industrial Informatics for the ***Special Section on Blockchain and AI enabled 5G Mobile Edge Computing***.

This paper is the authors' original work and has not been published nor has it been submitted simultaneously elsewhere. The submitted paper is of 8 pages.

The research contributions made in this work are as follows:

Secure and real-time data about goods in transit in supply chains needs bandwidth having capacity that is not fulfilled with the current infrastructure. Hence, 5G-enabled Internet of Things (IoT) in mobile edge computing is intended to substantially increase this capacity. To deal with this issue, we design a new efficient lightweight blockchain-enabled RFID-based authentication protocol for supply chains in 5G mobile edge computing environment, called LBRAPS. LBRAPS is based on bitwise exclusive-or (XOR), one-way cryptographic hash and bitwise rotation operations only. LBRAPS is shown to be secure against various attacks. Moreover, the simulation-based formal security verification using the broadly-accepted Automated Validation of Internet Security Protocols and Applications (AVISPA) tool assures that LBRAPS is secure. Finally, it is shown that LBRAPS has better trade-off among its security and functionality features, communication and computation costs as compared to those for existing protocols.

Thank you very much for your time to processing this matter.

With best regards,

Sincerely,

The Authors