A Novel Lightweight Block Cipher-Based Mutual Authentication Protocol for Constrained Environments

CUONG TRINH1, BAO HUYNH2, JAN LANSKY3, STANISLAVA MILDEOVA3, MASOUMEH SAFKHANI4, NASOUR BAGHERI5,6, SARU KUMARI7, AND MEHDI HOSSEINZADEH8,9

1Artificial Intelligence Laboratory, Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City 700000, Vietnam
2Faculty of Information Technology, Ho Chi Minh City University of Technology (HUTECH), Ho Chi Minh City 70000, Vietnam
3Department of Computer Science and Mathematics, Faculty of Economic Studies, University of Finance and Administration, 101 00 Prague, Czech Republic
4Computer Engineering Department, Shahid Rajaee Teacher Training University, Tehran 16788-15811, Iran
5Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran 16788-15811, Iran
6School of Computer Science (SCS), Institute for Research in Fundamental Sciences (IPM) at Farmanieh Campus, Tehran 19538-33511, Iran
7Department of Mathematics, Chaudhary Charan Singh University, Meerut 250004, India
8Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam
9Health Management and Economics Research Center, Iran University of Medical Sciences, Tehran 16668-87635, Iran

Corresponding author: Mehdi Hosseinzadeh (hosseinzadeh.m@iums.ac.ir)

This work was supported by the Institutional Research Project System Approach to selected ICT Trends under the University of Finance and Administration, Prague, under Grant 7429/2020/02.

ABSTRACT The communication security of constrained objects such as radio frequency identification (RFID) tags and wireless sensor network (WSN) is very challenging because it is not always possible to use the conventional on-the-shelf solutions for them, due to their limited available power and computational capabilities. To deal with this demand, many security protocols have been developed by the researchers so far. However, in many cases, the later analyses that have been carried out on these protocols have shown that they are vulnerable to one or few attacks, which could be enough to eliminate any application for such protocols. Following this direction, in this article, we analyze the security of four recent ultra-lightweight/lightweight protocols, by presenting important attacks including secret disclosure attack and desynchronization attack against them. To address the shortcoming of the previous protocols, we present a new security protocol based on lightweight block ciphers name it LBCbAP. In this protocol, we use CRAFT as the core security primitive. CRAFT is a tweakable block cipher which has been recently proposed and independent security analysis confirmed its security. Our detailed security analysis of LBCbAP, which is performed both informally and formally through the GNY logic and the Scyther tool, demonstrates its security against various types of attacks including secret disclosure and desynchronization attacks. The cost analysis of the designed protocol and comparison with the related lightweight protocols show that LBCbAP is cost efficient.

INDEX TERMS Internet of Things (IoT), mutual authentication, security analysis, lightweight protocol, GNY logic, scyther.

I. INTRODUCTION

Internet of Things (IoT) can be used to control a device through an Internet connection. IoT covers a wide range of applications at large and small scales. IoT technology has already been used in many applications including smart homes, smart cities, wearable technology, and smart cars. For example, a wearable gadget includes a series of sensors and uses special software to collect its user’s data; these gadgets ultimately analyze aggregated data such as health and fitness and report the results to the user. The IoT technology also has extensive application in vehicular system, smart grid and so on.

An important issue that has always been raised about the IoT is its security issue. Since this technology connects a lot of devices through the Internet, hacking them can have irreparable losses, such as losing sensitive personal and economic information. Therefore, a research challenge in this
field is to ensure users’ privacy and the confidentiality of their information by developing suitable software and hardware, to be used in Internet-based products. On the other hand, one of the solutions for implementing the IoT connection in the edge devices is the use of Radio Frequency Identification (RFID). Many solutions around the world generally use available components of RFID technology. RFID is used for security purposes mainly in the field of trade and warehouse logistics. Compared to other security technologies, RFID offers much greater potential than just for security itself, and so this technology is often deployed where monitoring is needed [1], e.g. patient medication safety [2]. However, RFID based devices are very constrained and cannot support conventional security solutions, which raises the security concerns of the related applications.

A. RELATED WORK AND MOTIVATION

IoT has brought the new potential to the Internet by enabling smarter communication between objects and man [3]. As stated in [3], these days the increasing amount of cyber attacks is a limit of IoT’s current paradigm, which is to connect any object anywhere. A factor that can help to expand the IoT is to ensure users of this technology in terms of maintaining their privacy and security, because if there is no proper security in this infrastructure, damage to IoT-based equipment, the possibility of losing personal information, the loss of privacy, and even the disclosure of economic and other data, are highly likely. Ronen et al. in [4] pointed out that if security is not taken into account in the IoT-based infrastructure, the technology will threaten the future of the world as a nuclear bomb. Hence, analyzing the proposed mechanisms from security aspect and designing new security modules that are compatible with IoT-based devices are also very important. Given that an IoT system can include many objects with limited resources, it requires special protocols to ensure that privacy and security of participants are guaranteed. Therefore, with the further development of IoT, its security concerns are expected to receive more attention. So far, several many protocols have been proposed to ensure IoT security, e.g. [5]–[9], however, most of them have failed in providing their security goals [10]–[13] and various attacks, such as the protocol’s secret values disclosure, desynchronization, traceability, impersonation were reported against those protocols. For example, Lee et al. proposed a new authenticated group key agreement protocol for imbalanced wireless networks [14]. However, later Cheng et al. showed that an active adversary can impersonate a mobile user efficiently [15]. Some other solutions proposed multi-factor authentication [16], [17] which may not be applicable in many RFID based applications, because in this case it is not practical to request the tagged object to do login operation by for example contributing its username and password. These attacks on the previous protocols resulted in a boost among protocol designers’ that are attempting to design protocols resilient against the published attacks and also compliant with more applications. However, there are still attacks against newly designed protocols. In addition, it would not be reasonable to trust any newly designed security protocol as far as it has not been enough analyzed by independent researchers. Hence, to address this demand, in this article we examine the security of four recent lightweight/ultra-lightweight protocols, to determine their pros and cons. It must be noted, although the proposed attacks are theoretical and have not been implemented against any real application, demonstrating different methods for attacking security protocols also helps the designers to consider those new lessons in their later designs to avoid past mistakes. The system model of our target application is represented in Figure 1. It is clear form this figure we consider the edge layer of a network where a reader communicating to tags over public channel. In this model, the adversary can control the communication between the reader and the tags passively or actively.

Two of the attacked protocols belong to ultra-lightweight protocols. This type of protocols, attempts to design a secure protocol for constrained environments with very simple functions such as bit-wise operations. There are many examples for such designing, e.g. SASI [18], RAPP [19] and SecLAP [20], and the later related third parties analysis [10], [21]–[23]. In the line of designing ultra-lightweight protocols, in [11], Wang et al. cryptanalyzed the Tewari and Gupta [24] protocol and also proposed an improved version of it. This protocol later analyzed by Khor and Sidorov [25], where they also proposed an improved protocol following the same designing paradigm. In this article, we consider the security of this improved protocol which has been proposed by Khor and Sidorov, and for simplicity, we call it KSP (stands for Khor and Sidorov protocol). We show that KSP is vulnerable to desynchronization attack and also secret disclosure attack.

As a new emerging technology, blockchain is believed to provide higher data protection, reliability, transparency, and lower management costs compared to a conventional...

---

**Figure 1.** The system model of the target applications.
centralized database. Hence, it could be a promising solution for large scale IoT systems. Targeting those benefits Sidorov et al. recently proposed an ultra-lightweight mutual authentication RFID protocol for blockchain-enabled supply chains [26]. Although they have claimed security against various attacks, we present an efficient secret disclosure attack on it. For the sake of simplicity, we call this protocol SOVNOKP.

Similar to SOVNOKP paradigm, Jangirala et al. also recently designed LBRAPS protocol [27] which is a lightweight blockchain-enabled RFID-based authentication and key agreement protocol. We also evaluate the security of this lightweight protocol.

Recently, Xiao proposed a mutual authentication protocol based on lightweight block cipher SKINNY [28] and named it LRSAS protocol [29]. Although they claimed optimal security, we present an efficient desynchronization attack against it which can be extended to traceability attack.

B. THE PAPER CONTRIBUTION

We can summarize the contributions of this article as below:

- We present the first third party analysis on two ultra-lightweight protocols, i.e. KSP and SOVNOKP, to the best of our knowledge. For example, we present efficient secret disclosure attack against these protocols. Following the proposed attacks in this article, one can extract whole secrets of those protocols with the complexity of only eavesdropping two sessions of each protocol between the tag and a legitimate reader, with negligible computational complexity. It is worth noting that an early draft of these attacks is also available in arxiv [30].
- We present the first third party analysis on two lightweight protocols, i.e. LBRAPS and LRSAS, to the best of our knowledge. In the case of LBRAPS, we present an efficient traceability attack and if the adversary could compromise a tag then it can also extract the secret of other tags. Our analysis for LRSAS is a very efficient desynchronization attack.
- We propose a new protocol and name it LBCbAP (Lightweight Block Cipher-based Authentication Protocol).
- Informally and also formally we provide the security proof of the proposed protocol using the GNY logic and the Scyther tool which show that LBCbAP provides desired security against different passive and active attacks such as secret disclosure attack, replay attack, and traceability attack.

C. PAPER ORGANIZATION

The rest of the paper is structured as follows: Section II is dedicated to a brief review of KSP, SOVNOKP, LBRAPS and LRSAS which are among recent proposals to provide security in the IoT/RFID environment. Our proposed secret disclosure attacks are applied to these protocols in Section III. We also proposed a secure protocol called LBCbAP in Section IV and its security proof is explained in Section V. The comparison of the proposed protocol with the related protocols is provided in Section VI. Finally, the paper is concluded in Section VII.

II. DESCRIPTION OF THE ATTACKED PROTOCOLS

To describe the protocols, we use the notations that are represented in Table 1. The notations are also used through the other parts of the paper.

A. KSP

Khor and Sidorov [25] proposed a new improved protocol which we call it KSP and runs as follows:

1) The reader $R$ starts the authentication phase of the protocol by sending $Hello$ and a random number $r$ to the tag $T_i$.
2) $T_i$, when receives the message, generates a random number $q$ and computes $M_1 = \text{Rot}(q \oplus r, wt(q))$, $M_2 = \text{IDS}^{\text{new}} \oplus r \oplus q$ and $M_3 = \text{Rot}(\text{IDS}^{\text{old}} \oplus \text{IDS}^{\text{new}}, wt(q))$ and sends $(M_1 || M_2 || M_3)$ to $R$.
3) Upon receipt of the message, $R$ does as follows:

- It does an exhaustive search to find $wt(q)$ to obtain a record of its database that satisfies the received $M_1$ and $M_2$, where $R$ will be able to find $\text{IDS}^{\text{old}}$ also, given $M_1$, to authenticate $T_i$.
- $R$ generates two new random numbers $m$ and $n$ and computes $M_4 = m \oplus n \oplus q$, $M_5 = \text{Rot}(n, wt(K))$.

### Table 1. Notations used in the protocols’ description.

| Symbol | Description |
|--------|-------------|
| $T$    | An RFID tag |
| $R$    | An RFID reader |
| $S$    | A supply chain node |
| $A$    | The adversary |
| $IDS$  | The pseudonym of the tag |
| $ID_T$ | The identifier of the tag |
| $ID_S$ | The identifier of the supply chain node |
| $K$    | The tag’s key |
| $m, n, q, r$ | The random numbers |
| $X_i$  | The $i^{th}$ bit of string $X$ |
| $wt(Y)$ | The hamming weight of $Y$ which equals with the number of 1’s in $Y$ |
| $\text{Rot}(X, Y)$ | The left rotation of $X$ using $Y$ |
| $\text{RRot}(X, Y)$ | The right rotation of $X$ using $Y$ |
| $\oplus$ | Exclusive-or operation |
| $H(\cdot)$ | Hash function |
| $E_K(\cdot)$ | Block cipher based encryption using the secret key $K$ |
| $E_{K, T}(\cdot)$ | Tweakeable block cipher based encryption using the secret key $K$ and the public tweak $T$ |
| $T'S$ | Timestamp |
and \( M_0 = Rot(Rot(K \oplus m, wt(n)), wt(K \oplus n)) \) and sends them to \( T_i \).

4) Once \( T_i \) receives the message, it:
   - Extracts \( n \) and \( m \) and verifies the messages integrity based on \( M_0 \) to authenticate \( R \).
   - If \( R \) has been authenticated, \( T_i \) and \( R \) will update their new parameters as \( IDS_{new} = Rot(IDS_{new} \oplus q, wt(n)) \oplus Rot(q, wt(m)) \) and \( K_{new} = Rot(K_{new} \oplus q \oplus n, wt(m)) \oplus Rot(m, wt(n)) \).

**B. SOVNOKP**

SOVNOKP [26] follows the KSP designing paradigm and it is based on almost similar components and runs as follows:

1) The reader \( R \) starts the authentication phase of the protocol by sending \textit{Hello} and a random number \( r \) to the tag \( T_i \).

2) \( T_i \), when receives the message, generates a random number \( q \) and computes \( M_1 = Rot(IDS \oplus r, wt(q)) \), and \( M_2 = Rot(IDS, wt(r)) \oplus Rot(K \oplus r, wt(q)) \) and sends \( \{M_1 || M_2\} \) to \( R \).

3) Upon reception of the message, \( R \) forwards \( r \| M_1 \| M_2 \) to the supply chain node.

4) Upon reception \( r \| M_1 \| M_2 \), the supply chain node does as follows:
   - It does an exhaustive search to find \( wt(q) \) to obtain a record of its database that satisfies the received \( M_1 \), where \( R \) will be able to find \( IDS \) and \( K \) to verify the received \( M_2 \) to authenticate \( T_i \).
   - Based on the \( H(IDS || K) \), it can check and track the product history together with permission level by reading the data from the blockchain. If the product has a correct history record in terms of ownership, timestamp, location, and product status, the supply chain node can authenticate \( T_i \).
   - Once \( T_i \) has been authorized, the supply chain node generates a random number \( m \) and computes \( M_3 \) and \( M_4 \) as:
     \[
     M_3 = \text{Rot}(r, wt(IDS) \oplus wt(K)) \oplus \text{Rot}(m, wt(K)) ,
     \]
     \[
     M_4 = \text{Rot}(m, wt(IDS)) \oplus \text{Rot}(r, wt(K)) \]
     and sends \( \{M_3 || M_4\} \) to \( R \).

5) \( R \) forwards the received \( \{M_3 || M_4\} \) to \( T_i \).

6) Once \( T_i \) receives the message, it:
   - Extracts \( m \) from \( M_4 \) and verifies the messages integrity based on \( M_4 \) to authenticate the reader/supply chain node.
   - If the reader/supply chain node has been authenticated and \( wt(m) \) is even, \( T_i \) will update its parameters as \( IDS_{new} = Rot(IDS \oplus K, wt(r)) \oplus Rot(K \oplus wt(q), wt(IDS)) \) and \( K_{new} = Rot(K, wt(r)) \oplus Rot(r \oplus q, wt(K)) \).

**C. LBRAPS**

LBRAPS [27] is a lightweight blockchain-enabled RFID-based authentication and key agreement protocol designed by Jangirala et al. This protocol uses hash function and also ultra-lightweight components such as rotation and XOR to provide a secure authentication between the tag, the reader and the supply chain node. In the initialization phase of the protocol, the reader and the tag are sharing their identifiers with the supply chain node and they consider their identifiers as the passwords. This protocol includes several steps but we only represent the first two steps of the authentication and key agreement phase of the protocol, which include a challenge-response communication between the reader and the tag, as follows:

1) The reader starts the authentication phase of the protocol by generating a random number \( r \) at time \( TS_R \) and computing \( M_1 = Rot(r \oplus ID_T \oplus TS_R \oplus ID_T) \) and \( M_2 = H(M_1 \| ID_T \| r) \) and sending \( \{M_1, M_2, TS_R\} \) to the tag.

2) The tag, when receives the message, verifies the validity of \( TS_R \), extracts \( r \) from \( M_1 \) and verifies the received \( M_2 \) to accept the received message. Next, at time \( TS_T \), it computes \( M_3 = Rot(r \oplus ID_S \oplus TS_T \oplus ID_T) \), \( M_4 = H(r \oplus ID_T \oplus Bal_{new}) \) and \( Auth_T = H(M_4 \| r \| M_3 \| ID_T \| TS_T) \) and sends \( \{M_4, Auth_T, M_3, TS_T\} \) to the reader.

3) We do not represent the rest of the protocol because it has no effect on the proposed analysis. \( Bal_{new} \) is the new balance related to the blockchain. Interested readers can find a full description of this protocol in [27].

**D. LRSAS**

LRSAS [29] is a recently proposed RFID authentication protocol based on a lightweight block cipher \textit{SKINNY} [28], designed by Xiao et al. In the initialization phase of this protocol, the tag’s identifier \( ID_i \), pseudonym \( IDS_i \) and key \( K_i \) are shared with the reader. The mutual authentication phase of the protocol is as follows:

1) The reader \( R \) starts the authentication phase by sending a request to the tag \( T_i \).

2) As the response, \( T_i \) sends its \( IDS_i \) to \( R \).

3) Given \( IDS_i \), \( R \) finds related \( (ID_i, K_i) \), generates a random number \( r \) and computes \( M_1 = IDS_i \oplus r \) and \( M_2 = E_{K_i}(IDS_i \oplus ID_i \oplus r) \) and sends \( \{M_1, M_2\} \) to \( T_i \).

4) The tag, when receives the message, extracts \( r \) as \( M_1 \oplus IDS_i \) and verifies whether \( M_2 = E_{K_i}(IDS_i \oplus ID_i \oplus r) \) to authenticate \( R \). Assuming \( R \) is authenticated, \( T_i \) computes \( M_3 = E_{K_i}(M_2 \oplus r) \) and sends it to \( R \).

5) \( R \) verifies whether \( M_3 = E_{K_i}(M_2 \oplus r) \) to authenticate \( T_i \). Next it assigns \( (IDS_i, K_i) \) to \( (IDS_i^{old}, K_i^{old}) \) and \( IDS_i^{new} = M_1 \) and uses the SKINNY tweakey schedule to compute \( K_i^{new} \) (its details have no effect on the proposed analysis). It also sends an \textit{OK} command to \( T_i \).

6) Once received the \textit{OK} command, \( T_i \) updates its pseudonym as \( IDS_i^{new} = M_1 \) and uses the SKINNY tweakey schedule to update its \( K_i^{new} \).

**III. SECURITY ANALYSIS OF KSP, SOVNOKP, LBRAPS AND LRSAS**

Although the designers of KSP and SOVNOKP argued their protocol’s security formally, it is easy to show that these
protocols are as vulnerable to different attacks as their predecessor rotation-based protocols. Designers of LBRAS claimed its security in the Canetti and Krawczyk’s adversary model (CK-adversary model) [31]. This protocol uses a hash function in calculating some messages, however, parts of the transferred messages are generated using ultra-lightweight operations including XOR and rotation. We use these parts to attack this protocol. Regarding LRSAS, although designers argued its security formally, we show that it does not provide the expected security.

A. SECRET DISCLOSURE ATTACK ON KSP

To analyze KSP, we follow the same adversary model as its designers, when they analyzed Wang et al.’s protocol [25, Sec.II, page 92], where “adversary is able to eavesdrop, intercept, block, and modify messages sent during communication between the reader and a tag”. Given the target tag, the attack procedure will be as follows:

1) The adversary eavesdrops a session of the protocol between the target tag and a legitimate reader, where, the reader generates an arbitrary nonce \( r \) and sends it to the tag, along \( \text{Hello} \).

2) The tag, when receives the message, generates a random number \( q \) and computes \( M_1 = \text{Rot}(q \oplus r, wt(q)) \), \( M_2 = IDS_{\text{new}} \oplus r \oplus q \) and \( M_3 = \text{Rot}(IDS_{\text{old}} \oplus IDS_{\text{new}}, wt(q)) \) and sends \( \{M_1|M_2|M_3\} \) to the reader, which is eavesdropped by the adversary.

3) Upon receipt of the message, the reader does as follows:
   - It does an exhaustive search to find \( wt(q) \) to obtain a record of its database that satisfies the received \( M_1 \) and \( M_2 \), where the reader will be able to find \( IDS_{\text{old}} \) also, given \( M_3 \), to authenticate the tag.
   - The reader generates two new random numbers \( m' \) and \( n' \) and computes \( M'_1 = m' \oplus n' \oplus q' \), \( M'_2 = \text{Rot}(n', wt(K)) \) and \( M'_3 = \text{Rot}(\text{Rot}(K \oplus m', wt(n')), wt(K \oplus n')) \) and sends them to the tag, which is eavesdropped by the adversary.

4) The adversary terminates the protocol.

5) Given that the tag has not received \( M_4 || M_5 || M_6 \), it will not update its records, i.e. \( IDS_{\text{new}}, K_{\text{new}}, IDS_{\text{old}} \) and \( K_{\text{old}} \).

6) The adversary waits for another session between the target tag and a legitimate reader, where, the reader generates an arbitrary nonce \( r' \) and sends it to the tag, along \( \text{Hello} \).

7) The tag, when receives the message, generates a random number \( q' \) and computes \( M'_1 = \text{Rot}(q' \oplus r', wt(q')) \), \( M'_2 = IDS_{\text{new}} \oplus r' \oplus q' \) and \( M'_3 = \text{Rot}(IDS_{\text{old}} \oplus IDS_{\text{new}}, wt(q')) \) and sends \( \{M'_1|M'_2|M'_3\} \) to the reader, which is eavesdropped by the adversary.

8) Upon receipt of the message, the reader does as follows:
   - It does an exhaustive search to find \( wt(q') \) to obtain a record of its database that satisfies the received \( M'_1 \) and \( M'_2 \), where the reader will be able to find \( IDS_{\text{old}} \) also, given \( M'_3 \), to authenticate the tag.
   - The reader generates two new random numbers \( m' \) and \( n' \) and computes \( M''_4 = m' \oplus n' \oplus q'' \), \( M''_5 = \text{Rot}(n', wt(K)) \) and \( M''_6 = \text{Rot}(\text{Rot}(K \oplus m', wt(n')), wt(K \oplus n')) \) and sends them to the tag, which is eavesdropped by the adversary.

9) Given that \( M_2 \oplus M'_2 = q \oplus r' \oplus q' \), the adversary finds two offset values \( c \) and \( c' \), respectively as \( wt(q) \) and \( wt(q') \), such that \( R\text{Rot}(M_1, c) \oplus R\text{Rot}(M'_1, c') = M_2 \oplus M'_2 \) and assigns \( R\text{Rot}(M_1, c) \oplus R\text{Rot}(M'_1, c') \) to \( q \oplus r \) and \( R\text{Rot}(M'_1, c') \) to \( q' \oplus r' \).

10) The adversary now extracts \( IDS_{\text{new}} = M_2 \oplus R\text{Rot}(M_1, c) \), \( q = R\text{Rot}(M_1, c) \oplus r \) and \( IDS_{\text{old}} = IDS_{\text{new}} \oplus R\text{Rot}(M_3, wt(q)) \).

11) The adversary finds three offset values \( c_1, c_2 \) and \( c_3 \), respectively as \( wt(K), wt(n) + wt(K \oplus n) \) and \( wt(n') + wt(K \oplus n') \), such that \( M_4 \oplus M'_4 \oplus q' = R\text{Rot}(M_5, c_1) \oplus R\text{Rot}(M'_5, c_1) \oplus R\text{Rot}(M_6, c_2) \oplus R\text{Rot}(M'_6, c_3) \).

12) The adversary assigns \( R\text{Rot}(M_5, c_1), R\text{Rot}(M'_5, c_1), R\text{Rot}(M_6, c_2) \) and \( R\text{Rot}(M'_6, c_3) \) respectively to \( n, n', K \oplus m \) and \( K \oplus m' \).

13) The adversary extracts \( m = M_4 \oplus R\text{Rot}(M_5, c_1) \oplus q \) and \( K = R\text{Rot}(M_6, c_2) \oplus m \).

Following the above attack, the adversary extracts whole shared parameters between the tag and the reader with the complexity of eavesdropping two sessions of the protocol, blocking a message, and doing polynomial computations. It is clear that \( M_2 \oplus M'_2 = q \oplus r' \oplus q' \oplus r' \). Now the adversary can also compute the updated values of \( IDS_{\text{new}} \) and \( K_{\text{new}} \) as \( IDS_{\text{new}} = R\text{Rot}(IDS_{\text{new}} \oplus q, wt(n)) \oplus R\text{Rot}(q, wt(m)) \) and \( K_{\text{new}} = R\text{Rot}(K_{\text{new}} \oplus q \oplus n, wt(m)) \oplus R\text{Rot}(m, wt(n)) \).

Remark 1: It should be noted the adversary can use other metrics to filter possible wrong guesses, e.g., \( c_1, c_2 \) and \( c_3 \) should be respectively equal to \( wt(K), wt(n) + wt(K \oplus n) \) and \( wt(n') + wt(K \oplus n') \).

Given that the adversary has whole secret parameters, applying any other attack will be trivial, e.g., reader/tag impersonation, desynchronization, and traceability attacks.

B. SECRET DISCLOSURE ATTACK ON SOVNOKP

Similar to KSP, to analyze SOVNOKP, we follow the same adversary model as the designers, when they analyzed Wang et al.’s protocol [26, Sec.IV, page 9], where “adversary has the capability to initiate communication with the reader and a tag and it is able to eavesdrop, intercept, block, and modify messages sent during communication between the reader and a tag. Moreover, number of unsuccessful attempts are limited to 5, otherwise the reader sends KILL command to the tag”. The parameter length is considered to be 64 bits. Given the target tag, the attack procedure, which is almost similar to the attack against KSP, will be as follows:

1) The adversary eavesdrops all transferred messages over a session of the protocol between the target tag and a legitimate reader, i.e., \( r, M_1, M_2, M_3, M_4 \) where, \( M_1 = \text{Rot}(IDS \oplus r, wt(q)), M_2 = \text{Rot}(IDS, wt(r)) \oplus \)
The adversary blocks $M_3 || M_4$, sent from the reader to the tag. Hence, the tag will not update its secrets, i.e., $IDS$ and $K$.

The adversary waits for another session between the target tag and a legitimate reader, where, the reader generates an arbitrary nonce $r'$ and sends it to the tag, along **Hello**.

The adversary again eavesdrops all transferred messages over a session of the protocol between the target tag and a legitimate reader, i.e., $r'$, $M'_1$, $M'_2$, $M'_3$, $M'_4$, where, $M'_1 = Rot(\text{IDS} \oplus r', \text{wt}(q'))$, $M'_2 = Rot(\text{IDS}, \text{wt}(r'))$, $M'_3 = Rot(r', \text{wt}(K))$, and $M'_4 = Rot(m', \text{wt}(\text{IDS})).$

5) Given that $\text{RRot}(M_1, \text{wt}(q')) \oplus \text{RRot}(M'_1, \text{wt}(q')) = IDS \oplus r \oplus IDS \oplus r' = r \oplus r'$ and $r$ and $r'$ are transferred over insecure channel, which is eavesdropped by the adversary, the adversary extracts $IDS$ and $K$ as follows:

   a) for $i \in \{0, 1, \ldots, 63\}$

   b) for $j \in \{0, 1, \ldots, 63\}$

   c) if $\text{RRot}(M_1, i) \oplus \text{RRot}(M'_1, j) = r \oplus r'$, returns $\text{RRot}(M_1, i) \oplus r$, and $j$ respectively as $IDS$, $\text{wt}(q)$, and $\text{wt}(q')$.

6) Given $r$, $r'$, $IDS$, $\text{wt}(q)$, and $\text{wt}(q')$, the adversary extracts $K$ as $\text{RRot}((M_2 \oplus \text{Rot}(\text{IDS}, \text{wt}(r')), \text{wt}(q')) \oplus r$.

7) The adversary can use other metrics to filter possible wrong guesses, e.g., $M'_2, M'_3, M'_4, M'_4$.

8) The adversary calculates the updated parameters as $IDS_{\text{new}} = \text{Rot}(\text{IDS} \oplus K, \text{wt}(r')) \oplus \text{Rot}(K \oplus \text{wt}(q'), \text{wt}(\text{IDS}))$ and $K_{\text{new}} = \text{Rot}(K, \text{wt}(r')) \oplus \text{Rot}(r \oplus \text{wt}(K))$.

Following the attack, the adversary extracts whole shared parameters between the tag and the reader with the complexity of eavesdropping two sessions of the protocol, blocking a message and doing polynomial computations. In this case also, since the adversary has whole secret parameters, applying any other attack will be trivial, e.g., reader/tag impersonation, desynchronization and traceability attacks.

### C. SECURITY ANALYSIS OF LBRAPS

Designers of **LBRAPS** claimed its security in the Canetti and Krawczyk’s adversary model (CK-adversary model) [31], which is a stronger adversary model compared to the Dolev-Yao (DY)-adversary model [32]. In the DY-adversary model, an adversary has full control over the transferred messages over the public channel and can eavesdrop, modify or delete the communicated messages among the various instances, or insert fake messages in between in communication. In the CK-adversary model, beside his/her abilities in the DY-adversary model, the adversary can also compromise the session states along with secret information, including secret keys. Thus, assuming the session states along with secret information are compromised in a particular session, this revealed information must not lead to compromise the secrecy of other parties [27].

1) **SECRET DISCLOSURE ATTACK ON LBRAPS**

In this section, we first extract a part of information related to the tag which can be used to trace it in a later session, in the DY-adversary model. Next, we extend the attack in the CK-adversary model to compromise the protocol’s security. Given a target tag $T_i$ which is communicating with the supply chain node $(S)$ through the reader $R$, the attack procedure will be as follows:

1) The reader generates a random number $r$ at time $TS_R$ and computes $M_1 = Rot(r \oplus ID_T \oplus TS_R, TS_R \oplus ID_T)$ and $M_2 = H(M_1 || ID_T || r)$ and sends $\{M_1, M_2, TS_R\}$ to the tag.

2) $A$ stores $\{M_1, M_2, TS_R\}$.

3) $T_i$ verifies the validity of $TS_R$, extracts $r$ from $M_1$ and verifies the received $M_2$ to accept the received message. Next, at time $TS_T$, it computes $M_3 = Rot(r \oplus ID_S \oplus TS_T, TS_T \oplus ID_T)$, $M_4 = H(r \oplus ID_T \oplus Bal_{\text{New}})$ and $\text{Auth}_R = H(M_3 || r || M_3 || ID_T || TS_T)$ and sends $\{M_4, \text{Auth}_R, M_3, TS_T\}$ to the reader.

4) $A$ stores $\{M_4, \text{Auth}_R, M_3, TS_T\}$ and sends a random value to $R$.

5) Hence, $R$ is expected to re-initiate the session by generating a random number $r'$ at time $TS_R$ and computing $M'_1 = Rot(r' \oplus ID_T \oplus TS'_R, TS'_R \oplus ID_T)$ and $M'_2 = H(M'_1 || ID_T || r')$ and sending $\{M'_1, M'_2, TS'_R\}$ to the tag.

6) $A$ stores $\{M'_1, M'_2, TS'_R\}$.

7) $T_i$ verifies the validity of $TS'_R$, extracts $r'$ from $M'_1$ and verifies the received $M'_2$ to accept the received message. Next, at time $TS'_T$, it computes $M'_3 = Rot(r' \oplus ID_S \oplus TS'_T, TS'_T \oplus ID_T)$, $M'_4 = H(r' \oplus ID_T \oplus Bal_{\text{New}}')$ and $\text{Auth}'_R = H(M'_4 || r' || M'_4 || ID_T || TS'_T)$ and sends $\{M'_4, \text{Auth}'_R, M'_4, TS'_T\}$ to the reader.

8) $A$ stores $\{M'_4, \text{Auth}'_R, M'_4, TS'_T\}$ and does as follows:

   a) Creates two lists $L_1$ and $L_2$.

   b) For $x =: wt(TS_R \oplus ID_T) \in \{0, 1, \ldots, 63\}$

      - For $y =: wt(TS'_T \oplus ID_T) \in \{0, 1, \ldots, 63\}$

        - Stores $\text{RRot}(M_1, x) \oplus \text{RRot}(M_3, y) \oplus TS_R \oplus TS'_T \in L_1$ as a candidate for $ID_T \oplus IDS_R$.

   c) For $x =: wt(TS'_R \oplus ID_T) \in \{0, 1, \ldots, 63\}$

      - For $y =: wt(TS'_T \oplus ID_T) \in \{0, 1, \ldots, 63\}$

        - Stores $\text{RRot}(M'_1, x) \oplus \text{RRot}(M'_3, y) \oplus TS'_R \oplus TS'_T \in L_2$ as a candidate for $ID_T \oplus IDS_R$.

   d) Returns the matching value in $L_1$ and $L_2$ as the value of $ID_T \oplus IDS_R$.

Following the above attack, which is in the DY-adversary model, the adversary could extract $ID_T \oplus IDS_R$ which is expected to be confidential. This information is enough to trace $T_i$, as long as it is in communication with the same $S$. Next assume to follow the CK-adversary model, which has been used by the designers of **LBRAPS**. We assume
that the adversary compromised a tag $T_i$ and achieved its identifier $ID_i$. Given the above attack, the adversary can also extract $ID_S$. On the other hand, given $ID_S$, the adversary can extract $ID_T$ for any other tag $T_j$, applying the given attack. It shows that in the CK-adversary model it is possible to retrieve the identifier of any tag, which contradicts the designers’ claim. Given the identifier of the tag, which is its password also, it is also possible to extract $r$ and other shared values in each session, which have been considered secure in this protocol.

**D. DESYNCHRONIZATION ATTACK ON LRSAS**

In this section, we present our analysis on LRSAS. The main drawback of this scheme is lack of session randomizing by $T_i$. Hence, it is possible to desynchronize the tag and the reader following the procedure described in [10] against other protocols. Therefore, we just present a brief description of the attack as follows:

1. The reader $R$ starts the authentication phase of LRSAS by sending a request to the tag $T_i$.
2. As the response, $T_i$ sends its $IDS_i$ to $R$.
3. Given $IDS_i$, $R$ finds related $(ID_i, K_i)$, generates a random number $r$ and computes $M_1^* = IDS_i ⊕ r$ and $M_2^* = E_{K_i}(IDS_i ⊕ ID_i ⊕ r)$ and sends $(M_1^*, M_2^*)$ to $T_i$.
4. The adversary $A$ stores $(M_1, M_2)$ and blocks them. Hence, the protocol is terminated and the reader is expected to start another session.
5. $R$ starts the authentication phase by sending a request to $T_i$.
6. As the response, $T_i$ sends its $IDS_i$ to $R$.
7. Given $IDS_i$, $R$ finds related $(ID_i, K_i)$, generates a random number $r^*$ and computes $M_1^* = IDS_i ⊕ r^*$ and $M_2^* = E_{K_i}(IDS_i ⊕ ID_i ⊕ r^*)$ and sends $(M_1^*, M_2^*)$ to $T_i$.
8. The tag, when receives the message, extracts $r^*$ as $M_1^* ⊕ IDS_i$ and verifies whether $M_2^* = E_{K_i}(IDS_i ⊕ ID_i ⊕ r^*)$ to authenticate $R$, which is authenticated and $T_i$ computes $M_3^* = E_{K_i}(M_2^* ⊕ r^*)$ and sends it to $R$.
9. $R$ verifies whether $M_2^* = E_{K_i}(M_2^* ⊕ r^*)$ and authenticates $T_i$. Next it assigns $(IDS_i, K_i)$ to $(IDS_i, K_i)$ and $IDS_i' = M_1^*$ and uses the SKINNY tweakey schedule to compute $K_i'$ and sends an OK command to $T_i$, which is blocked by $A$ and $T_i$ does not update its record.
10. $A$ impersonates $R$ by sending a request to $T_i$.
11. As the response, $T_i$ sends its $IDS_i$.
12. $A$ returns the stored $(M_1, M_2)$.
13. The tag, when receives the message, extracts $r'$ as $M_1 ⊕ IDS_i$ and verifies whether $M_2 = E_{K_i}(IDS_i ⊕ ID_i ⊕ r')$ to authenticate $R$ which is $A$. Clearly, $T_i$ accepts the message and computes $M_3 = E_{K_i}(M_2 ⊕ r')$ and sends it to $R$.
14. $A$ sends an OK command to $T_i$.
15. $T_i$ updates its pseudonym as $IDS_i' = M_1$ and uses the SKINNY tweakey schedule to update its $K_i'$.

Following this attack, the new pseudonym of $T_i$, i.e. $IDS_i' = M_1$, does not match any of the reader’s records with a high probability. Hence, the adversary successfully desynchronized the tag and the reader. Hereafter, the tag’s response on each request remains identical, i.e. $IDS_i'^{new} = M_1$, which is enough to trace it also.

**IV. LBCbAP: A SECURE LIGHTWEIGHT PROTOCOL**

In the previous sections, we pointed out the security pitfalls of four ultra-lightweight protocols, which beside other related studies demonstrates the shortcoming of this approach to design a secure protocol. On the other hand, many lightweight block ciphers have been proposed in recent years. The lightweight block ciphers CRAFT [33], GIFT [34], SKINNY [28] and SIMON [35] or some proposed modes of operation to guarantee both the confidentiality and message integrity such as IAR-CTR and IAR-CFB [36] are some examples. It is worth noting that most of those block ciphers can be implemented even in passive RFID tags, based on their reported implementation results. For example, the unprotected version of CRAFT, which has been recently proposed, requires only 949 Gate Equivalents (GE) [33, P.30,Table 6]. Besides, recently NIST is organizing a competition for lightweight cryptography [37] which already has announced its second-round candidates also. Hence, thanks to the recent advances in lightweight ciphers, to provide a secure protocol for the constrained environments, such as RFID tags and IoT edge devices, we propose a new authentication protocol based on lightweight block ciphers. In this section, we propose LBCbAP, a lightweight block cipher-based authentication protocol. As the main building block, we use CRAFT block cipher [33], which has been proposed in FSE’19 by Beierle et al. CRAFT is a tweakable block cipher, which maps a 64-bit plaintext to a 64-bit cipher using a 128-bit secret key $K = K_0 || K_1$ and a 64-bit tweak $T$. It is worth noting that a tweakable block cipher [38] not only accepts the usual plaintext and secret key but also a third input which is known as the “tweak”, which serves much the same purpose that an initialization vector does for CBC (Cipher Block Chaining) mode of block ciphers or stream ciphers. The best-known distinguisher for this cipher in single tweak mode covers 14 rounds [39] out of 32 rounds. It shows that, although the cipher is very lightweight, it provides good security also. Details of this cipher can be found in [33]. It is worth noting that the current distinguishers are in the known/chosen plain text modes which are not practical in many protocols because the adversary has no access to the input plaintext in general. Hence, it should be possible to achieve desired security using a reduced round version of the block ciphers.

LBCbAP has oracle access to CRAFT as $C = E_{K,T}(P)$ or $P = E_{K,T}^{-1}(C)$, where $E$, $E^{-1}$, $P$, $C$, $T$ and $K$ respectively denote encryption, decryption, plaintext, ciphertext, public tweak and secret key. LBCbAP includes three parties, a tag $T_i$, a reader $R$ and a trusted database. The channel between the tag and the reader is public while the channel between the reader and the database is secure. We assume that in the initialization phase, the database stores
the tuple \(<ID_i, K_i, IDS_j>\) as the identifier, the shared secret key and the pseudonyms of \(T_i\) in the tag securely. It could be equivalent to programming the tag in the factory. These parameters are also stored in the trusted database as \(<ID_i, (K_i^{new}, IDS_i^{new}), (K_i^{old}, IDS_i^{old})>\), in step \((K_i^{new}, IDS_i^{new}) = (K_i^{old}, IDS_i^{old})\). In addition, acceptable threshold for challenge-response time is also set for \(R\) and \(T_i\) respectively as \(\Delta T_R\) and \(\Delta T_T\). The mutual authentication phase of LBCbAP between \(R\) and \(T_i\) is depicted in Figure 2 and runs as follows:

1) At the time \(T_{SR}\), the reader \(R\) starts the authentication phase by generating a random number \(r\) and sending it with the \(Hello\) message to the tag.

2) The tag \(T_i\), when receives the \(Hello\) message and \(r\), generates a random number \(q\) and computes \(M_1 = E_{K_i,q}(ID_i \oplus r)\) and sends \((IDS_i, M_1, q)\) to \(R\). It also runs a timer \(TS_T\) initiated by zero.

3) At time \(T_{SR}\), \(R\) receives \((IDS_i, M_1, q)\). It first verifies whether \(T_{SR} - TS_R \leq \Delta T_R\) to avoid relay attacks; otherwise terminates the protocol. Next, given \(IDS_i, R\) finds related secrets \(<K_i, ID_i>\) (looks at the new records first and then at the old records) and verifies the received \(M_1\) to authenticate \(T_i\).

4) \(R\) stores the used \(IDS_i, K_i\) as \((IDS_i^{old}, K_i^{old})\), calculates \(M_2||IDS_i^{new}||K_i^{new} = E_{K_i,q}(IDS_i^{new}||(ID_i \oplus r))\), updates \((IDS_i^{new}, K_i^{new})\), where \((K_i^{new}) = (K_i)\). Accordingly and sends \(M_2\) to \(T_i\).

5) Once received \(M_2\), \(T_i\) verifies whether \(T_{ST} \leq \Delta T_T\) to avoid relay attacks; otherwise terminates the protocol. Next, it computes \(M_2||IDS_i^{new}||K_i^{new} = E_{K_i,q}(IDS_i^{new}||(ID_i \oplus r))\). If the received \(M_2\) is valid then \(T_i\) authenticates \(R\) and sets \((K_i^{new}) = (K_i)\).

The mutual authentication phase of LBCbAP runs as follows:

If time is \(T_{SR}'\), verifies \(T_{SR}' - TS_R \leq \Delta T_R\); based on \(IDS_i\), finds \(<K_i, ID_i>\) (looks at the new records first and then the old records); verifies \(M_1\) to authenticate \(T_i\); stores the used \((IDS_i, K_i)\) as \((IDS_i^{old}, K_i^{old})\); calculates \(M_2||IDS_i^{new}||K_i^{new} = E_{K_i,q}(IDS_i^{new}||(ID_i \oplus r))\); updates \((IDS_i^{new}, K_i^{new})\), where \((K_i^{new}) = (K_i)\).

FIGURE 2. Mutual authentication phase of LBCbAP.

V. SECURITY ANALYSIS OF LBCbAP

Possible approaches to provide security proof of a cryptographic protocol are divided into formal and informal categories. Informal methods of security proof are based on analytical knowledge and are mostly based on cryptanalyst creativity, while formal methods, which are themselves divided into two categories of manual methods and automated methods, are either model checking based or Theorem prover based or logic-based such as the logic of belief. Tools such as Proverif [40], AVISPA [41] and Scyther [42] belong to automated formal methods while methods such as BAN logic [43], SVO [44, 45] and GNY logic [46] are among manual formal methods. In this section, beside informal security analysis, we prove the correctness of our proposal both informally and formally through the GNY logic and the Scyther tool.

A. INFORMAL SECURITY ANALYSIS

Our adversary model follows CK-adversary model, where the adversary can control the channel and if it compromises a tag then the security of the other tags should not be affected.

1) TRACEABILITY ATTACK

To perform the traceability attack on a protocol, the adversary tries to connect the transferred messages over different sessions or link them with the protocol’s party. In the LBCbAP, transferred messages are \(r, Hello, IDS_i, M_1, q\), where \(M_1 = E_{K_i,q}(ID_i \oplus r)\) and \(M_2||IDS_i^{new}||K_i^{new} = E_{K_i,q}(IDS_i^{new}||(ID_i \oplus r))\). \(Hello\) is a meaningless message.
for the adversary to be used for traceability. \( r \) and \( q \) are also fresh nonces that the adversary cannot control both. \( M_1 \) and \( M_2 \) are also randomized by \( r \) and \( q \) and sound random to \( \mathcal{A} \), assuming the used cipher is secure. The only parameter which could be used to trace the tag is \( \text{IDS}_i \). However, after a successful session of the protocol, this parameter is also updated and randomized by \( r \) and \( q \). It is worth noting that in symmetric-crypto based protocols, if we aim to keep scalability, this type of traceability seems to be unavoidable.

It is possible to mask \( \text{IDS}_i \) for example by redefining \( M_1 \) as \( M_1 = \text{IDS}_i \oplus E_{K_{i,q}}(ID_i \oplus r) \). However, then to find the tag, the reader should do the same calculation for any entry in its database and the complexity will be \( O(N) \), where \( N \) is the number of tags with valid records, while in the current approach the complexity of finding the tag is \( O(1) \). Therefore, if the user traceability is very important, we suggest to use the revised \( M_1 \), i.e. \( M_1 = \text{IDS}_i \oplus E_{K_{i,q}}(ID_i \oplus r) \), in the cost of losing scalability.

2) SECRET DISCLOSURE ATTACK
In LBCbAP, sensitive data are included in \( M_1 \) and \( M_2 \) that are calculated using a secure symmetric cipher. Hence, the adversary cannot extract an information regarding \( ID_i \) and \( K_i \) in polynomial time. Hence, LBCbAP is secure against secret disclosure attack.

3) REPLAY ATTACK
Given that, in LBCbAP both the tag and the reader are contributing to the randomness of the transferred messages \( M_1 \) and \( M_2 \), the adversary cannot replay the eavesdropped messages from an old session in a later one. Hence, LBCbAP is resistant against the replay attack.

4) IMPERSONATION ATTACK
Given that it is not practical to do a replay attack, to impersonate \( \mathcal{T}_i \) the adversary \( \mathcal{A} \) should provide a valid response \( M_1 = E_{K_{i,q}}(ID_i \oplus r) \) to the given challenge \( r \) by \( \mathcal{R} \). However, \( \mathcal{A} \) does not have access to \( ID_i \) and \( K_i \) to compute the proper \( M_1 \). In addition, it cannot return a previously stored value exclude that \( r \) has already been used by \( \mathcal{R} \), which has a small probability in our setting. On the other hand, to impersonate \( \mathcal{R} \) the adversary \( \mathcal{A} \) should provide a valid response \( M_2 \) to the given challenge \( q \) by \( \mathcal{T}_i \). However, given that \( \mathcal{A} \) does not have access to \( ID_i \) and \( K_i \) it has no chance to return correct \( M_2 \) directly. In addition, it cannot return a previously stored value exclude that \( q \) has already been used by \( \mathcal{T}_i \), which has a small probability in our setting. Therefore, LBCbAP is resistant to the impersonation attack.

5) DESYNCHRONIZATION ATTACK
Desynchronization attack is an important attack against any protocol which can contradict the availability service [47]. To desynchronize \( \mathcal{T}_i \), the adversary should do a successful impersonation attack or compromise the integrity of the transferred message without being detected by the receiver. However, neither of them is practical against LBCbAP because it is resistant against impersonation attack, and the integrity of the messages is checked on the receiver side. For example, if the adversary modifies any bit of \( M_1 \) then the output of the decrypted message will be random and the possibility of passing \( \mathcal{R} \)’s verification will be \( 2^{-|r|} \), where \( |r| \) denotes the bit length of \( r \). A similar argument is valid for \( M_2 \). However, the adversary can block or modify \( M_2 \) such that \( \mathcal{R} \) is updating its records but \( \mathcal{T}_i \) is not but in this way the current records in tag also exist in \( \mathcal{R} \), as the old record. Hence, \( \mathcal{R} \) authenticates \( \mathcal{T}_i \) using its old records. Hence, LBCbAP is secure against desynchronization attack.

6) FORWARD SECRECY
As a requirement in the CK-adversary model security, assuming that the adversary compromised \( \mathcal{T}_i \) in time \( t \) and achieved its secrets at this time, i.e. \( ID_i \), \( K_i \) and \( \text{IDS}_i \), it should not be capable to link them with the previously transferred data or compromise that data security. After each successful session of LBCbAP, \( \text{IDS}_i \) and \( K_i \) are updated as \( M_2||\text{IDS}_i^{\text{new}}||K_i^{\text{new}} = E_{K_{i,q}}(\text{IDS}_i||K_0)\ |\ | (ID_i \oplus r) \) and \( (K_0)_{\text{old}}^{\text{new}} = (K_1)_{\text{old}} \), where \( r \) and \( q \) are session dependent nonces. On the other hand, given \( M_2||\text{IDS}_i^{\text{new}}||K_i^{\text{new}}, \text{IDS}_i, r \) and \( q \), the adversary cannot determine whole \( K_i \) if the used cipher is secure in known plaintext mode. Hence, the adversary cannot determine the key of the previous session given the key of the current session. Therefore, it will not be able to compromise the forward secrecy of the protocol and we conclude that LBCbAP is secure in the CK-adversary model.

7) MAN IN THE MIDDLE ATTACK
To provide security against man in the middle relay attacks, \( \mathcal{R} \) and \( \mathcal{T}_i \) are counting the message travel time. In addition, any modification in the transferred messages are detected by the receiver of the message because \( M_1 \) and \( M_2 \) have built in integrity checking mechanisms, e.g. any modification in \( M_1 \) leads to random output and pass the \( \mathcal{R} \) checking of the correct \( r \), as its challenge, with the probability of \( 2^{-|r|} \). Hence, LBCbAP is resistant against man in the middle attacks.

B. FORMAL SECURITY ANALYSIS
In this section, we formally prove LBCbAP satisfies the security goals of the protocol by using GNY logic [46]. The security proof of LBCbAP through GNY logic is explained as follows:

1) Expression of LBCbAP messages as mathematical equations: in this step, LBCbAP is explained as below:
   \[
   M_1 : \quad \mathcal{R} \rightarrow \mathcal{T}_i : \text{Hello, } r;
   \]
   \[
   M_2 : \quad \mathcal{T}_i \rightarrow \mathcal{R} : \text{IDS}_i, M_1 = E_{K_{i,q}}(ID_i \oplus r) \text{ and } q;
   \]
   \[
   M_3 : \quad \mathcal{R} \rightarrow \mathcal{T}_i : \text{M}_2 \text{ where } M_2||\text{IDS}_i^{\text{new}}||K_i^{\text{new}} = E_{K_{i,q}}(\text{IDS}_i||K_0)\ |\ | (ID_i \oplus r).
   \]

2) Expression of LBCbAP messages based on GNY logic notations: in this step, we express the protocol messages using GNY logic notations which are represented in Table 2 as below. It must be noted that because
of the ease of analysis, the messages transmitted in the
diagram are written as a function of their inputs.
M1: \( T_i \leftarrow \text{Hello}, *r \);
M2: \( R \leftarrow \text{IDS}_i, *q, \{ F_3(ID_i, r, q) \}_K \);
M3: \( T_i \leftarrow \{ F_4(IDS_i, (K_0), ID_i, r, q) \}_K \);

3) Idealizing LBCbAP’s messages: in this step, we
remove the messages that do not increase confidence,
as follows:
IM1: \( R \leftarrow \{ F_3(ID_i, r, q) \} \rightarrow \exists R \equiv R \leftarrow T_i \mid K_i \); IM2: \( T_i \leftarrow \{ F_4(IDS_i, (K_0), ID_i, r, q) \} \rightarrow T_i \equiv T_i \mid K_i \).

4) Define security assumptions and security goals of
LBCbAP: in this step, we express the assumptions and
also the security goals of LBCbAP as follows:

Protocol assumptions:
A1: \( R \leftarrow \text{IDS}_i \);
A2: \( R \leftarrow r \);
A3: \( R \leftarrow \phi(r) \);
A4: \( T_i \leftarrow q \);
A5: \( T_i \leftarrow q \);
A6: \( T_i \leftarrow \phi(q) \);
A7: \( T_i \leftarrow T_i \mid K_i \rightarrow R \);
A8: \( R \leftarrow R \leftarrow T_i \);

A9: \( T_i \leftarrow T_i \mid K_i \rightarrow R \);
A10: \( R \leftarrow R \leftarrow ID_3 \rightarrow T_i ;
A11: \( T_i \leftarrow T_i \mid ID_3 \rightarrow R \);
A12: \( R \leftarrow R \leftarrow ID_3 \rightarrow T_i ;
A13: \( T_i \leftarrow K_i ;
A14: \( R \leftarrow K_i ;

Protocol security goals:
G1: \( R \leftarrow T_i \mid K_i \rightarrow F_3(ID_i, r, q) \);
G2: \( T_i \leftarrow R \leftarrow F_4(IDS_i, (K_0), ID_i, r, q) \);

5) Retrieving LBCbAP security goals: in this step, we
obtained the protocol security goals using the messages
and assumptions of the protocol according to the
GNY logic rules as follows:

Given A1, based on F1 we get D1: \( R \rightarrow T_i \leftarrow T_i \mid K_i \rightarrow T_i \).
With D1 and A14 and based on F2 we deduce that
D2: \( R \rightarrow R \leftarrow T_i \leftarrow F_3(ID_i, r, q) ;
Considering A3 and based on R1, we get D3: \( R \leftarrow \phi(F_3(ID_i, r, q) ;
Given IM1, A14, A8, D3 and D2, based on I1 rule of
GNY logic, we get D4: \( R \leftarrow R \leftarrow T_i \leftarrow F_4(IDS_i, (K_0), ID_i, r, q) ;
which is the G1 security goal.
With A4, then based on F1 rule of GNY logic, we get
D5: \( T_i \leftarrow T_i \mid K_i \rightarrow T_i \);
Based on the previous result, i.e. $D5$ and $A13$, and based on $F2$ we get $D6 : T_i | \equiv (\text{Id}([F4(IDSi), (K0)_i, ID_i, r, q])_k)$;

Given $A6$ and $R1$ rule we get $D7 : T_i | \equiv \phi(F4(IDSi), (K0)_i, ID_i, r, q)$;

Following $IM2$, $A13$, $A7$, $D7$ and $D6$ and based on $I1$ rule we get $D8 : T_i | \equiv \sim F4(IDSi, (K0)_i, ID_i, r, q)$ which is the $G2$ security goal.

As you can see, the security objectives of the protocol are achieved through $D4$ and $D8$ which indicate that the adversary has no control over the values transmitted in the protocol and also the adversary cannot change the protocol messages, without the receiver realizing, or cannot impersonate the identity of one protocol’s party to participate in the authentication protocol in such a way that his/her identity is confirmed as a legal protocol’s party.

C. Formal VERIFICATION OF LBCbAP security

The Scyther tool is a Python based on-the-shelf tool which is used to verify the security of cryptographic protocols [42]. To analyze a protocol using this tool, the target protocol should be modeled using the Security Protocol Description Language (SPDL) and then the Scyther tool is applied to verify the security claims of the protocol.

To verify the security of LBCbAP, we modeled it using SPDL as described in Figure 3. In our modeling, we defined two entities tag and reader. The verification result using the Scyther tool is depicted in Figure 4. This result shows that the Scyther tool could not find any attack within bounds on LBCbAP, which proves our security claims for LBCbAP.

VI. COMPARISON

In this section, we compare LBCbAP with KSP, SOVNOKP, LBRAPS and LRSAS from security and efficiency perspectives. Through our analysis we target 64-bit security, although the secret key length could be higher. Hence, the input/output length of each call of the block cipher is 64 bits, the output length of $H(.)$ is 128 bits, the random numbers and the length of other parameters such as $ID$ and $IDS$ are also 64 bits. The length of timestamp is 32 bits.

A. SECURITY COMPARISON

In subsection III-B and subsection III-A, we respectively described the full secret disclosure attack on SOVNOKP and KSP protocols. It is clear given the secret parameters of a protocol, it will be easy to do any other attack, e.g. impersonation a protocol’s party or trace it. Hence, as it is depicted in Table 3, these protocols suffer from any attack in that comparison table. The proposed security analysis of LBRAPS in subsection III-C could discover a part of the protocol’s secret parameters which could be used to trace the tag. However, it is not enough to impersonate it. In addition, the protocol does not update any parameter. Hence, it does not suffer from desynchronization attack.

In section V, we provided detailed security analysis of LBCbAP against different attacks and have shown informally and also formally that it provides desired security against different attacks. For example, in subsections V-A5, V-A4, V-A2 and V-A1 we argued the security of LBCbAP respectively against desynchronization attack, impersonation attack, secret disclosure attack and traceability attack. A summary of the security comparison of LBCbAP versus other protocols are provided in Table 3. It can be seen LBCbAP provides better security compared to the
M sends TS. It also runs a timer LBCbAP some it is secure. Through a session of protocols, that channel in some protocols is insecure while in communication from the reader to the database for all protocols. In the analysis, we only report the costs of the tags, because the analysis of the computational cost is lower than LBCbAP compared to lighter protocols is the cost we should pay to provide the desired security. It should be noted to reduce the tag’s cost we designed messages such that it only needs encryption function and does not need access to the decryption function while LRSAS needs both encryption and decryption functions.

TABLE 3. Security comparisons between LBCbAP and analyzed protocols.

| Claim      | Status | Comments      |
|------------|--------|---------------|
| LBCbAP     | R      | Secret ID    |
| LBCbAP,R1  | R      | No attacks within bounds. |
| LBCbAP,R2  | R      | No attacks within bounds. |
| LBCbAP,R3  | R      | No attacks within bounds. |
| LBCbAP,R4  | R      | No attacks within bounds. |
| LBCbAP,R5  | R      | No attacks within bounds. |
| LBCbAP,R6  | R      | No attacks within bounds. |
| LRSAS [29] | R      | No attacks within bounds. |
| LBCbAP     | T      | Secret Key |
| LBCbAP,T1  | T      | No attacks within bounds. |
| LBRAPS     | T      | No attacks within bounds. |
| LBRAPS,T2  | T      | No attacks within bounds. |
| LBSAP      | T      | No attacks within bounds. |
| LBCbAP     | R      | Weak Syndrome |
| LBCbAP,T4  | R      | No attacks within bounds. |
| LBCbAP,T5  | R      | No attacks within bounds. |
| LBCbAP,T6  | R      | No attacks within bounds. |

analyzed protocols. While these results are almost expected for KSP and SOVNOKP, because they are ultra-lightweight, however, LBRAPS and LRSAS could do better because they use hash function or block cipher. On the other hand, it shows that LBCbAP, based on a lightweight block cipher, is secure against the attacks presented in this article and other known active and passive attacks.

B. COMPUTATIONAL COST COMPARISON

TABLE 4. Computational cost comparisons.

| Protocol | # $\oplus/$ | # $R/rot(X,Y)$ | # $E_K(X)$ | # $H(\cdot)$ |
|----------|-------------|----------------|------------|--------------|
| KSP [25] | 12          | 9              | -          | -            |
| SOVNOKP [26] | 11          | 11             | -          | -            |
| LBRAPS [27] | 14          | 4              | -          | 5            |
| LRSAS [29] | 4           | -              | 2          | -            |
| LBCbAP    | 1           | 4              | -          | -            |

FIGURE 5. LBCbAP versus related protocols, computational cost comparison.

FIGURE 4. The proposed protocol verification results using the Scyther tool.

TABLE 3. Security comparisons between LBCbAP and analyzed protocols.  
1 G1: Resistance against desynchronization attack,  
2 G2: Resistance against impersonation attack.  
3 G3: Resistance against secret disclosure attack.  
4 G4: Resistance against traceability attack.

| Protocol | G1 | G2 | G3 | G4 |
|----------|----|----|----|----|
| KSP [25] | x  | x  | x  | x  |
| SOVNOKP [26] | x  | x  | x  | x  |
| LBRAPS [27] | x  | x  | x  | x  |
| LRSAS [29] | x  | x  | x  | x  |
| LBCbAP    | x  | x  | x  | x  |

Table 4 represents the computational comparison of LBCbAP and analyzed protocols. Through a session of protocol, the tag first computes $M_2\|IDS_{new}(K_1_{new}) = E_{K,q}(IDS_{new}(K_0))(ID_i \oplus r)$. If the received $M_2$ is valid then $T_i$ authenticates $R$ and sets ($K_0_{new} = (K_1)$, $K_i = (K_1)_{new}||K_0_{new}$ and $IDS_i = IDS_{new}^+$; otherwise terminates the protocol. It is clear $M_1$ costs a call to the block cipher because the bit length of its input is 64 bits. However, the bit-length of the input message to the block cipher during the computation of $M_2||IDS_{new}^+(K_1_{new})$ is 64 + 64 + 64 bits, which requires 3 calls to the block cipher. Besides it requires to compute $ID_i \oplus r$, which costs a call to XOR function. Hence, the computational cost of LBCbAP is 4 calls to the block cipher plus a call to the XOR function.

Table 4 shows that the computational complexity of LBCbAP is much lower than LBRAPS but higher than KSP and SOVNOKP. However, those two protocols, i.e. KSP and SOVNOKP, are ultra-lightweight and do not provide enough security. LRSAS computational cost is lower than LBCbAP. However, all those protocols are insecure. Hence, the higher cost of LBCbAP compared to lighter protocols is the cost we should pay to provide the desired security. It should be noted to reduce the tag’s cost we designed messages such that it only needs encryption function and does not need access to the decryption function while LRSAS needs both encryption and decryption functions.

C. COMMUNICATION COST COMPARISON

Table 5 represents the communication comparison of LBCbAP and analyzed protocols. To provide a fair comparison, we only counted the communication cost from the tag to the reader and vice versa, i.e. we eliminated possible communication from the reader to the database for all protocols. Through a session of protocol, the tag first computes $M_1$ and sends $IDS_i, M_1, q$ to $R$, where its bit-length is 64 + 64 + 64. It also runs a timer $TS_i$ initiated by zero. At the first, $R$ starts the authentication phase by generating a random
number \( r \) and sending it with the Hello message to the tag. Eliminating the bit length of Hello because it is a wake-up signal, in this stage the reader transferred 64 bits and the tag received 64 bits. Next, the tag sends \( \{IDS, M_1, q\} \) with the length of 64+64+64 to the reader. Hence, in this stage, the tag transferred 192 bits and the reader receives 192 bits. Finally, the reader sends \( M_2 \) to the tag that is 64 bits. Hence, in this stage, the reader sends 64 bits and the tag receives 64 bits.

All in all, the tag has received 128 bits of data and transferred 192 bits of data and the reader has received 192 bits of data and transferred 128 bits of data. Following this computation and the same computation for other protocols that are provided in Table 5, also depicted in Figure 6, LRSAS has the lowest communication cost among the compared protocols and after that LB\(_b\)AP, SO\(_{VNOKP}\), K\(_S\)P and LB\(_R\)APS respectively provide less communication costs. Among these protocols, LB\(_R\)APS has the highest communication cost.

### VII. CONCLUSION

In this article, we have shown that four recent proposed security protocols for IoT and RFID systems (i.e. K\(_S\)P, SO\(_{VNOKP}\), LB\(_R\)APS and LRSAS) are vulnerable against secret disclosure attack and desynchronization attack. We presented attacks with the complexity of mostly eavesdropping only two sessions of each protocol and negligible computational complexity. For an attack on LB\(_R\)APS which is presented in the Canetti and Krawczyk’s adversary model, we need to compromise a tag also. Besides, we presented LB\(_b\)AP, a new secure protocol based on lightweight block ciphers, supported by formal and informal security proof.

The presented attacks on K\(_S\)P and SO\(_{VNOKP}\) again highlight that when we aim to design a secure protocol, we must not only focus on boosting the performance by limiting the operations at the expense of weakening the very basic principles of security, this weak approach will only lead to secret disclosure attacks as we have presented. On the other hand, the presented attacks on LB\(_R\)APS and LRSAS show that employing a secure component such as one-way hash functions or block ciphers in a protocol does not guarantee its security, if the transferred messages are not designed properly.

Some directions for future works are also presented here. For each analyzed protocol, the proposed attacks are the first attacks against each of these protocols to the best of our knowledge. Although the proposed attack against each protocol is enough efficient to rule out any real application for that protocol, however, it does not rule out further analysis of those protocols. For example, in the case of LRSAS, we are only able to present desynchronization attack. Although this attack is enough to contradict the designers’ security claim but it does not mean that some one cannot propose a secret disclosure attack for example by finding a weakness in the used block cipher (SKINNY). Hence, we suggest to investigate the possible application of the previous cryptanalysis of that block cipher, e.g. [48]–[51], on the security of the protocol. While we analyzed the protocol in an ideal cipher model, this type of analysis has not received enough attention in literature and we believe it provides better understanding of the protocol’s security in a real application. In LB\(_b\)AP, we have not considered a separate database and assumed that the reader is enough safe. However, in some applications such as mobile readers, this assumption does not hold. In such an application it is not reasonable to store the tag secrets directly in the reader, due to the risk of stolen reader attack, which can be addressed in future works. Besides, if we consider a system with multiple readers, we need a database to support all those readers to keep the updated key and IDS of the tags; otherwise, it will be desynchronized in the current form. Hence, designing an RFID system based on LB\(_b\)AP that supports multiple readers is also an interesting target for future works. In that research, the challenges such as how the reader initializes its database, can the communication be restored if the reader’s database is lost and how the tag can communicate with multiple readers and stay synchronous and

| Protocol | \( T_i-R \) | \( T_i-T \) | \( R-R \) | \( R-T \) |
|----------|------------|------------|--------|--------|
| K\(_S\)P [25] | 256 | 192 | 192 | 256 |
| SO\(_{VNOKP}\) [26] | 192 | 128 | 128 | 192 |
| LB\(_R\)APS [27] | 448 | 352 | 352 | 448 |
| LRSAS [29] | 128 | 128 | 128 | 128 |
| LB\(_b\)AP | 128 | 192 | 192 | 128 |
anonymous could be answered. Last but not least, the proposed protocol is still under evaluation (not yet implemented in real environment) and still there is a need to be further analyzed by the community before possibly deploying in any operational environment.

REFERENCES

[1] D. Brodani, “A case study of RFID technology application for protection of computing technology against theft and Users’ tendency to bypass the protection,” Acta Inf. Pragensia, vol. 8, no. 1, pp. 4–17, Jul. 2019.

[2] C. Jin, C. Xu, X. Zhang, and F. Li, “A secure ECC-based RFID mutual authentication protocol to enhance patient medication safety,” J. Med. Syst., vol. 40, no. 1, p. 12, Jan. 2016.

[3] P. P. Ray, “A survey on Internet of Things architectures,” J. King Saud Univ.-Comput. Inf. Sci., vol. 30, no. 3, pp. 291–319, 2018.

[4] E. Ronen, A. Shamir, A.-O. Weingarten, and C. O’Flynn, “IoT goes nuclear: Creating a ZigBee chain reaction,” in Proc. IEEE Symp. Secur. Privacy (SP), May 2017, pp. 195–212.

[5] B. Mukherjee, R. L. Neupane, and P. Calyam, “End-to-End IoT security middleware for cloud-fog communication,” in Proc. IEEE 4th Int. Conf. Cyber Secur. Cloud Comput. (CSCloud), Jun. 2017, pp. 151–156.

[6] M. Taylor, D. Reilly, and B. Lemerpreur, “An access control management protocol for Internet of Things devices,” Netw. Secur., vol. 2017, no. 7, pp. 11–17, Jul. 2017.

[7] Z. Wang, “A privacy-preserving and accountable authentication protocol for IoT end-devices with weaker identity,” Future Gener. Comput. Syst., vol. 82, pp. 342–348, May 2018.

[8] A. Serhrouchni and P. Minet, “A lightweight mutual authentication protocol for the IoT,” J. ICMTW, vol. 425, 2017, p. 3.

[9] C.-T. Li, C.-Y. Weng, and C.-C. Lee, “A secure RFID tag authentication protocol with privacy preserving in telecare medicine information system,” J. Med. Syst., vol. 39, no. 9, p. 77, Aug. 2015.

[10] M. Safkhani and N. Bagheri, “Generalized desynchronization attack on UMAP: Application to RCIA, KMAP, SLAP and SASI protocols,” in Proc. IACR, 2016, p. 905.

[11] K.-H. Wang, C.-M. Chen, W. Fang, and T.-Y. Wu, “On the security of a new ultra-lightweight authentication protocol in IoT environment for RFID tags,” J. Supercomput., vol. 74, no. 1, pp. 65–70, Jan. 2018.

[12] M. Sarvahatla, M. Giri, and C. S. Vorugunti, “Cryptanalysis of a biometric-based user authentication scheme for heterogeneous wireless sensor networks,” in Proc. 7th Int. Conf. Contemp. Comput. (IC3), Aug. 2014, pp. 312–317.

[13] C.-C. Lee, M.-S. Hwang, and I.-E. Liao, “Security enhancement on a new authentication scheme with anonymity for wireless environments,” IEEE Trans. Ind. Electron., vol. 53, no. 5, pp. 1683–1687, Oct. 2006.

[14] C.-C. Lee, T.-H. Lin, and C.-S. Tsai, “A new authenticated group key agreement in a mobile environment,” Ann. Telecommun. Commun., vol. 64, nos. 11–12, pp. 735–744, Dec. 2009.

[15] Q.-F. Cheng, C.-G. Ma, and F.-S. Wei, “Analysis and improvement of a new authenticated group key agreement in a mobile environment,” Ann. Telecommun. Commun., vol. 66, nos. 5–6, pp. 331–337, Jun. 2011.

[16] C.-T. Chen and C.-C. Lee, “A two-factor authentication scheme with anonymity for multi-server environments,” Secure Commun. Netw., vol. 8, no. 8, pp. 1608–1625, May 2015.

[17] A. Omotov, S. Bezzeatev, N. Maktalo, S. Andreev, T. Mikkonen, and Y. Koucheryavy, “Multi-factor authentication: A survey,” Cryptography, vol. 2, no. 1, p. 1, Jan. 2018.

[18] E. Y.-C. Chien, “SASSE: A new ultra-lightweight RFID authentication protocol providing strong authentication and strong integrity,” IEEE Trans. Dependable Secure Comput., vol. 4, no. 4, pp. 337–340, Oct. 2007.

[19] Y. Tian, G. Chen, and J. Li, “A new ultra-lightweight RFID authentication protocol with permutation,” IEEE Commun. Lett., vol. 16, no. 5, pp. 702–705, May 2012.

[20] S. F. Aghili, H. Mala, P. Kalijary, and M. Conti, “SecLAP: Secure and lightweight RFID authentication protocol for medical IoT,” Future Gener. Comput. Syst., vol. 101, pp. 621–634, Dec. 2019.

[21] G. Avoine, X. Carpent, and B. Martin, “Privacy-friendly synchronized ultra-lightweight authentication protocols in the storm,” J. Netw. Comput. Appl., vol. 35, no. 2, pp. 826–843, Mar. 2012.

[22] M. Xiao, W. Li, X. Zhong, K. Yang, and J. Chen, “Formal analysis and improvement on ultra-lightweight mutual AuthenticationProtocols of RFID,” Chin. J. Electron., vol. 28, no. 5, pp. 1025–1032, Sep. 2019.
MA SOUMEH SAFKHANI received the Ph.D. degree in electrical engineering from the Iran University of Science and Technology, in 2012, with the security analysis of RFID protocols as her major field. She is currently an Assistant Professor with the Computer Engineering Department, Shahid Rajaee Teacher Training University, Tehran, Iran. Her current research interests include the security analysis of lightweight and ultra-lightweight protocols, targeting constrained environments, such as RFID, the IoT, VANET, and WSN. She is the author/coauthor of over 50 technical articles in information security and cryptography in major international journals and conferences.

NASOUR BAGHERI received the M.S. and Ph.D. degrees in electrical engineering from the Iran University of Science and Technology (IUST), Tehran, Iran, in 2002 and 2010, respectively. He is currently an Associate Professor with the Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran. He is also a part-time Researcher with the Institute for Research in Fundamental Sciences. He is the author of more than 100 articles in information security and cryptography. His research interests include cryptography, more precisely, designing, and analysis of symmetric schemes, such as lightweight ciphers, e.g., block ciphers, hash functions, and authenticated encryption schemes, cryptographic protocols for constrained environment, such as RFID tags and the IoT edge devices and hardware security, e.g., the security of symmetric schemes against side-channel attacks, such as fault injection and power analysis.

SARU KUMARI received the Ph.D. degree in mathematics from Chaudhary Charan Singh University, Meerut, India, in 2012. She is currently an Assistant Professor with the Department of Mathematics, Chaudhary Charan Singh University. She has published more than 133 research articles in reputed International journals and conferences, including 115 publications in SCI-indexed journals. Her current research interests include information security and applied cryptography. She is currently a Technical Program Committee Member for many International conferences. She has served as a Lead/Guest Editor of four special issues in SCI journals of Elsevier, Springer, and Wiley. She is on the Editorial Board of more than 12 journals of international repute, including seven SCI journals.

MEHDI HOSSEINZADEH received the B.S. degree in computer hardware engineering from Islamic Azad University, Dezful Branch, Iran, in 2003, and the M.Sc. and the Ph.D. degrees in computer system architecture from the Science and Research Branch, Islamic Azad University, Tehran, Iran, in 2005 and 2008, respectively. He is currently an Associate Professor with the Iran University of Medical Sciences (IUMS), Tehran. He is the author/coauthor of more than 120 publications in technical journals and conferences. His research interests include SDN, information technology, data mining, big data analytics, e-commerce, e-marketing, and social networks.

---

**C. Trinh et al.: Novel Lightweight Block Cipher-Based Mutual Authentication Protocol**

---

**CUONG TRINH** received the M.Sc. degree in computer science from Military Technical Academy, Vietnam, in 2014. He is currently pursuing the Ph.D. degree in informatics, communication technology, and applied mathematics with the Technical University of Ostrava, Czech Republic. His research interests include data mining, parallel computing, network infrastructure, and network security. In addition, he has over ten years of experience in design and analysis of symmetric schemes, such as lightweight ciphers, and applications in various fields. His current research interests include data mining, privacy-preserving, big data analytics, parallel computing, social networks, network infrastructure, and network security. In addition, he has more than 120 publications in Information Security and Cryptography.

**BAO HUYNH** received the M.Sc. degree in computer science from the Posts and Telecommunications Institute of Technology, Ho Chi Minh City, Vietnam, in 2011, and the Ph.D. degree in computer science from the VSB-Technical University of Ostrava, Czech Republic, in 2017. Since that time, he has been continuously researched and developed new algorithms that are helpful for real applications in various fields. His research interests include data mining, privacy-preserving, big data analytics, parallel computing, social networks, network infrastructure, and network security. In addition, he has more than 120 publications in Information Security and Cryptography.

**JAN LANSKY** received the M.S. and Ph.D. degrees in computer science software systems from Charles University, Prague, Czech Republic, in 2005 and 2009, respectively. He has been a Professor with the Department of Computer Science and Mathematics, Faculty of Economic Studies, University of Finance and Administration, Prague, Czech Republic, since March 2009. He has been a Head of the Department, since September 2014. His research interests include cryptocurrencies, text compression, and databases.

**STANISLAVA MILDEOVA** graduated from the University of Economics, at Prague. She is currently an Associate Professor and the Deputy Head of the Department of Informatics and Mathematics, Faculty of Economic Studies, University of Finance and Administration Prague, Czech Republic, since 2017. In her work, she focuses on applied informatics and systems science with a focus on systems dynamics. She has been the Editor-In-Chief of the Scopus Journal Acta Informatica Pragensia.