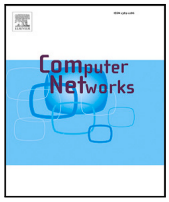
Computer Networks 252 (2024) 110678

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet



Anonymous and reliable ultralightweight RFID-enabled authentication scheme for IoT systems in cloud computing

Mohd Shariq [a,b,*], Mauro Conti [c], Karan Singh [d], Chhagan Lal [e], Ashok Kumar Das [f,g], Shehzad Ashraf Chaudhry [h,i], Mehedi Masud [j]

[a] Department of Computer Science and Engineering, Sharda University, Greater Noida 201310, India
[b] Department of Information Systems & Security, College of Information Technology, United Arab Emirates University, Al Ain, Abu Dhabi UAE
[c] Department of Mathematics, University of Padua, 35131 Padua, Italy
[d] School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi 110067, India
[e] Department of Information Security and Communication Technology, NTNU, Norway
[f] Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad, 500032, India
[g] Department of Computer Science and Engineering, College of Informatics, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul, 02841, South Korea
[h] Department of Computer Science and Information Technology, College of Engineering, Abu Dhabi University, Abu Dhabi, United Arab Emirates
[i] Department of Software Engineering, Faculty of Engineering and Architecture, Nisantasi University, Istanbul, Turkey
[j] Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

ARTICLE INFO

Keywords:
Big data
Authentication
IoT
RFID
Cloud computing
Scyther tool

ABSTRACT

With the continuing growth in various research domains such as Artificial Intelligence (AI), Machine Learning (ML), Big Data Analytics (BDA), and Cloud Computing (CC), the Internet of Things (IoT) has become a popular technology nowadays. It provides a virtual interaction between physical objects and the cyber world over the Internet without human intervention. The IoT devices are embedded with sensors, software, or some other useful technologies for connecting, sharing, and exchanging data with other devices. Among recent technologies impacting human lives, Radio Frequency IDentification (RFID) has become a core identification technology that can be integrated into the IoT environments which connect billions of things or objects. However, an RFID system has two major issues: (i) an adversary can tamper or intercept the sensitive information of the RFID tags, which may cause forgery and privacy problems, and (ii) RFID tags have limited computational power capability which makes it challenging to use current security solutions. To deal with these issues, we propose an anonymous and reliable ultralightweight RFID-enabled authentication scheme (namely AnonR^2AS) for IoT systems in a cloud computing environment. AnonR^2AS integrates bitwise exclusive-OR, left–right rotations, and ultralightweight $half\text{-}flip$ operation, to reduce computational overheads on tag. The AnonR^2AS provides stronger security (by preventing several attacks) and improves performance concerning low computational, communication, and storage costs. Also, it preserves information privacy and tags untraceability property by using Vaudenay privacy model. The Scyther simulation tool verification has been performed for its formal security analysis. The performance analysis ensures our AnonR^2AS scheme is preferable for low-cost RFID systems.

1. Introduction

In the era of ubiquitous computing, IoT technologies have become more prominent and are rapidly growing everywhere. IoT networks are composed of various sensor devices such as smart objects, RFID tags, and many other sensor-enabled devices [1]. All these devices are equipped with the capability to collect, store, and communicate data to other devices deployed worldwide. Thus, IoT has become an imperative technology in context with the current inter-connected world [2]. At present, IoT is being widely used in a huge number of real-life applications such as Industrial IoT, telemedicine, connected vehicles, e-healthcare, industrial Internet, smart cities, smart homes, smart retail, smart grid, smart farming, smart supply chain, so forth [3–5].

As a key component of IoT systems, the group of Radio Frequency IDentification (RFID) technologies are referred to as Automatic Identification and Data Capturing (AIDC). In particular, RFID technology is

* Corresponding author.
E-mail addresses: shariq99ansari@gmail.com, mohammad.shariq@uaeu.ac.ae (M. Shariq), mauro.conti@unipd.it (M. Conti), karancs12@gmail.com (K. Singh), chhagan.lal@ntnu.no (C. Lal), iitkgp.ashok.das@iiit.ac.in (A.K. Das), shehzad.ashraf@adu.ac.ae (S.A. Chaudhry), mmasud@tu.edu.sa (M. Masud).

https://doi.org/10.1016/j.comnet.2024.110678
Received 8 February 2024; Received in revised form 19 July 2024; Accepted 26 July 2024
Available online 29 July 2024

becoming a contactless and wireless technique used to automatically identify and track the tagged objects via radio frequency signals [6–8].

Typically, an RFID system comprises an RFID reader, RFID tags, and a backend database. The RFID tag is a small device that consists of an antenna, an Integrated Circuit (IC), and dedicated hardware for cryptographic primitives that transmit data to the RFID reader. The tags generally store the sensitive information associated with the objects and communicate it to the RFID reader for authentication purposes. Generally, RFID tags have very limited computational capability and storage capacity when compared to the reader and server. The RFID tags consist of active, semi-active, and passive tags [9]. The active tags contain a radio transceiver and a small battery. These tags can communicate directly with the reader. The semi-active tags contain a small battery and get charged from it [10]. The passive tags get power and charge wirelessly with the help of a reader. The RFID reader consists of a radio transmitter/receiver, a memory unit, and a control unit.

To accomplish mutual authentication, RFID reader enables the tag and server to exchange messages among themselves. Usually, the computing capacity of the RFID reader is higher in comparison to the RFID tag [11,12]. The backend server is always considered to be a completely trusted entity, it stores all the secret information related to RFID tags in its database. The server computes the tag's validity using stored secret information in the database. The RFID system enables the identification of numerous tag-attached objects concurrently. Cloud computing addresses the limitations and performance of the physical or backend servers in the IoT environment when a huge amount of data is being generated in an RFID-based IoT system. The cloud platform has several promising advantages in contrast to physical servers such as high scalability, data storage, and low-cost services [13,14].

The two important features of RFID systems, i.e., data processing and robustness, can greatly enhance the performance of the cloud server using cloud computing technologies [15,16]. Besides, the cloud platform is rapidly evolving and can be deployed in RFID systems, where robustness ensures the reliability and stability of the system. In the cloud platform, all the data can be collected and processed by the sensors which solves the serious problems involved in RFID systems such as data loss and data delay [17]. During the system setup, some secrets are pre-shared between the tag and server to achieve secure authentication over an insecure communication channel. The data exchanged wirelessly over the insecure channel may lead to adversaries being able to easily modify, tamper, steal, or even intercept the data [18]. The channel is assumed to be secure between RFID reader and server because some pre-shared secret keys and some other security mechanisms are established between them.

1.1. Problem statement and motivation

There exists a variety of RFID-based authentication schemes for safeguarding RFID systems from various security and privacy attacks. These schemes are typically based on traditional lightweight cryptographic primitives, for instance, symmetric key encryption and one-way hash functions. The RFID system considers two communication channels for transmitting messages: secure (between reader and cloud) and insecure (between tag and reader). The adversary may infer or forge the RFID tag's information, and compromise the tag's privacy over an insecure channel. In addition to this, the transmitted messages may be tampered, injected, replayed, and/or intercepted over the communication channel by an adversary. These issues pose a threat to the security of RFID systems. To fix the aforementioned security flaws, we propose an anonymous and reliable ultralightweight RFID-enabled authentication scheme (AnonR^{2}AS) for IoT systems in a cloud computing environment. In addition to security and privacy, RFID tags contain limited computational capability and storage compared to reader and server. Existing cloud-assisted RFID authentication scheme(s) may suffer from the high computational overhead of the tags. To reduce the computational overhead, we exploit simple bitwise exclusive-OR, left–right rotations, and ultralightweight $half\text{-}flip$ operations in our proposed scheme.

1.2. Security requirements for an RFID system

With regards to an RFID system, this section briefly introduces some of the common security requirements defined as [19,20]:

- **Information privacy** $\mathcal{A}$ can access the secret information stored in the tags.
- **Eavesdropping attack** All the messages are being transmitted between tags and readers over an insecure communication channel. $\mathcal{A}$ may easily listen or eavesdrop. Hence, adversaries can obtain secret information or confidential data from eavesdropped messages.
- **Location tracking** $\mathcal{A}$ may try to track the location of the tag-embedded objects.
- **Replay attack** First, $\mathcal{A}$ eavesdrops on the transmitted messages to obtain the sensitive information of tags or readers in one authentication session. Next, using this information the adversary can replay in the future and can get more information regarding the tag.
- **Impersonation attack** First, $\mathcal{A}$ masquerades as an authorized or trusted tag (or reader), then it can impersonate the tag (or reader) using information collected from the previous communication session.
- **De-synchronization attack** $\mathcal{A}$ can intercept the important transmitted messages and make the changes in the tag's secret key stored in the backend server in an authentication session. However, the tag's secret key stored in the tag cannot be the same as the tag's secret key which is stored in the database of the backend server.
- **Disclosure attack** $\mathcal{A}$ can slightly modify the messages from the reader and transmit these messages to the tag to verify the correctness of modifications.

1.3. Objectives

The key objectives of our scheme are as follows.

- ensure the tag anonymity, and mutual authentication between $\mathcal{T}$ and $\mathcal{CS}$.
- ensures several known privacy features including tag untraceability, tag information, and tag location privacy.
- ensures security features including confidentiality and scalability.
- resists several known attacks including impersonation and replay attacks.
- minimizes communication and computational overhead of tags.

1.4. Key contribution

After an extensive study of existing cloud-assisted RFID authentication schemes, we have pointed out a variety of security, privacy, and performance issues of these schemes (refer to Section 2). Keeping aforesaid issues in view, we propose an anonymous and reliable RFID-enabled authentication scheme (AnonR^{2}AS). In particular, this paper has the following contributions.

- An anonymous and reliable RFID-enabled authentication scheme (AnonR^{2}AS) is proposed for IoT systems in a cloud computing environment. It exploits bitwise XOR, left rotate operation, and an ultralightweight $half\text{-}flip$ operation to encrypt data instead of a one-way hash which consumes more computation time.
- Our AnonR^{2}AS scheme achieves information privacy and tags untraceability property. During authentication process, the useful plaintext information of tags cannot be obtained by the cloud server, which provides privacy protection for tags. The informal security analysis of AnonR^{2}AS shows it is resistant to several known attacks like replay and impersonation.

- We provide an implementation of AnonR^{2}AS. The performance comparison is performed with the other similar schemes which demonstrates that AnonR^{2}AS greatly reduces the computational overhead of tags while supporting the above-mentioned security and privacy guarantees.

1.5. Roadmap of the paper

The rest of the paper is arranged as follows. The related work is presented in Section 2. Section 3 presents brief details about the materials and methods used in our proposed scheme. Section 4 presents the RFID system model and adversarial model, respectively. Next, Section 5 details our proposed AnonR^{2}AS scheme. Sections Section 6 present the adversary classes and proofs of information privacy-preserving and tags untraceability using Vaudenay privacy model. Further, Section 7 presents informal and formal security analysis, followed by performance analysis. The concluding remark and future directions are presented in Section 8.

2. Related work

Over the last few years, a variety of RFID authentication schemes using cloud computing have been proposed by many researchers for safeguarding RFID systems from several known security attacks. Table 1 discusses some existing cloud-based RFID authentication schemes with their respective class, cryptographic primitive, pros and cons.

Xie et al. [21] introduced a privacy-preserving cloud-centered RFID authentication scheme. In this scheme, a Virtual Private Network (VPN) agency is deployed to organize the secure backend channels. The scheme considers the cloud database as an encrypted hash table and employs simple XOR ($\oplus$), concatenation ($\|$) operations, hash function $h(\cdot)$, Pseudo-Random Number Generator ($PRNG$), and encryption $E(\cdot)_k$/ decryption $D(\cdot)_k$ function by using a symmetric algorithm. However, the study by Abughazalah et al. [22] highlighted vulnerabilities such as reader impersonation and tag location tracking attacks in Xie's scheme [21].

Abughazalah et al. [22] first showed the security weaknesses of Xie et al.'s [21] scheme which was vulnerable to tag location tracking and reader impersonation attacks, and privacy invasion of the tag's data. Furthermore, they presented an improved version of Xie's scheme to ensure security and privacy for the tags using simple XOR ($\oplus$), hash function $h(\cdot)$, $PRNG(\cdot)$, and performing symmetric operations. The scheme claims to achieve tag anonymity, tag location privacy, replay, tag/reader impersonation, and de-synchronization attacks. However, Surekha B et al. [23] showed the scheme [22] cannot withstand the tag location privacy property.

Xiao et al. [24] presented a cloud-centered RFID authentication approach over an insecure channel. The scheme uses simple bitwise XOR ($\oplus$), hash function $h(\cdot), PRNG(\cdot)$, and performs $E(\cdot)_k/D(\cdot)_k$ by employing a symmetric algorithm through the shared master keys. It preserves the tag privacy feature and protects data secrecy. In addition, complicated hash operations are performed on the tag in each authentication session, which results in a high computational cost. Also, the scheme cannot realize the identity authentication feature.

Fan et al. [25] presented an efficient and secure cloud-based RFID scheme. The scheme integrates a hash function $h(\cdot)$ which is used for data encryption between tags and readers. The protocol uses two timestamps for the tag and the reader for updating the authentication information which resists synchronization attacks. The protocol can withstand forgery, replay, and tag-tracking attacks.

Fan et al. [26] proposed a secure privacy-preserving and efficient cloud-based RFID scheme. The scheme uses simple XOR, permutation $Per(\cdot,\cdot)$, and left rotation $Rot(\cdot,\cdot)$ operations. The use of timestamps updates the tag's secret information as well as ensures the message's freshness. The scheme resists replay and de-synchronization attacks. However, Adeli et al. [27] mentioned out that Fan's scheme is insecure against forward–backward security, anonymity, impersonation, secret disclosure, replay, traceability, man-in-the-middle, and de-synchronization attacks.

Fan et al. [28] proposed a lightweight RFID protocol for IoT environment. They used left rotation $Rot(\cdot,\cdot)$, cross $Cro(\cdot,\cdot)$, $PRNG(\cdot)$, bitwise exclusive-OR, and concatenation ($\|$) operations. The protocol resists some typical attacks including replay, Denial-of-Service (DoS), and synchronization attacks. However, Aghili et al. [29] presented an improvement of the Fan's protocol [28] called SecLAP. Later on, Safkhani et al. [30] showed traceability, reader impersonation, and full and partial secret disclosure attacks in [29].

Anand et al. [31] proposed a reliable and efficient ultralightweight RFID authentication scheme for patients' medication safety called ER^{2}AS. The scheme uses bitwise XOR, circular left/right rotations, and reformation operation. The scheme resists several known attacks such as impersonation, disclosure, replay, and de-synchronization attacks. However, Ebrahimi and Nikooghadam [32] showed that the scheme is insecure against insider and impersonation attacks.

Shariq and Singh [33] introduced a lightweight RFID authentication protocol using vector-space-based approach. The protocol ensures several known security features including tag anonymity, mutual authentication, forward secrecy, etc. The protocol can also withstand disclosure, replay, and tag location tracking attacks. However, the authors do not provide any formal security analysis.

3. Preliminaries

We briefly provide an overview of circular left and right rotations in this section. To achieve higher security, our proposed AnonR^{2}AS scheme employs simple XOR, circular left rotation, and ultralightweight $half\text{-}flip$ operations. The key parameters used with their respective symbols and abbreviations are shown in Table 2.

3.1. Rank and nullity

The number of 1's and 0's present in a string is called rank and nullity. Suppose $\mathcal{X}/\mathcal{Y}$ is a given string, the rank, and nullity are defined as follows.

$rank(\mathcal{X}/\mathcal{Y})$ = Number of 1's presented in string $\mathcal{X}/\mathcal{Y}$

$nullity(\mathcal{X}/\mathcal{Y})$ = Number of 0's presented in string $\mathcal{X}/\mathcal{Y}$.

3.2. Circular rotation operations

The two operations namely circular left and right rotations used in our AnonR^{2}AS scheme. The rotation operation is generally denoted as $\mathcal{X}, \mathcal{Y}$. $Rot_l(\mathcal{X}, \mathcal{Y})$ denotes the left rotation where string $\mathcal{X}$ left rotates by $rank(\mathcal{Y})(mod\ L)$ bits. While, $Rot_l(\mathcal{X}, \mathcal{Y})$ denotes the right rotation where string $\mathcal{X}$ rotates by $rank(\mathcal{Y})(mod\ L)$ bits. The outcomes of the left and right rotations might be $\mathcal{X}$ or $\mathcal{Y}$ with the probability $\frac{1}{L}$, where L denotes the bit length of string $\mathcal{X}$ or $\mathcal{Y}$.

- **Left and right rotate operation** Suppose $\mathcal{X}$ and $\mathcal{Y}$ are two 8-bit strings:

  $\mathcal{X} = 10111001, \mathcal{Y} = 01011000,$

  Then, the left rotate operation on $\mathcal{X}$ with $rank(\mathcal{Y})$ can be computed as $Rot_l(\mathcal{X}, \mathcal{Y})$ =String $\mathcal{X}$ is left rotated by $rank(\mathcal{Y})$. The rank of string $\mathcal{Y}$ is 3, i.e., $rank(\mathcal{Y}) = 3$. Now, $rank(\mathcal{Y})(mod\ 8) = 3\ mod\ 8 = 3$.
  Therefore, $Rot_l(\mathcal{X}, \mathcal{Y}) = Rot_l(\mathcal{X}, 3) = 11001101$.
  Similarly, the right rotate operation on $\mathcal{X}$ with $rank(\mathcal{Y})$ can be computed as, $Rot_r(\mathcal{X}, \mathcal{Y})$ =String $\mathcal{X}$ is left rotated by $rank(\mathcal{Y})$. The rank of string $\mathcal{Y}$ is 3 i.e., $rank(\mathcal{Y}) = 3$. Now, $rank(\mathcal{Y})(mod\ 8) = 3\ mod\ 8 = 3$.
  Therefore, $Rot_r(\mathcal{X}, \mathcal{Y}) = Rot_r(\mathcal{X}, 3) = 00110111$.

Table 1
Comparison among various cloud-based RFID authentication schemes.

Reference	Class	Cryptographic primitive	Pros	Cons
Ref [21]	* Lightweight	* Bitwise XOR * Concatenation * Hash function * Pseudo-Random Number Generator * Encryption/decryption	* The scheme preserves tag and reader privacy against database keepers.	* Insecure against reader impersonation and tag location tracking attacks.
Ref [22]	* Lightweight	* Bitwise XOR * Concatenation * Hash function * Pseudo-Random Number Generator * Encryption/decryption	* The scheme achieves tag anonymity, tag location privacy, replay, tag/reader impersonation, and de-synchronization attacks.	* Insecure against tag location privacy property.
Ref [24]	* Lightweight	* Bitwise XOR * Hash function * PRNG * Encryption and decryption by employing a symmetric algorithms	* The scheme preserves tag privacy feature and protects data secrecy.	* The scheme suffers from high computational costs and also cannot realize the identity authentication feature.
Ref [25]	* Lightweight	* Hash function * Concatenation * Timestamps for tag and reader Encryption * Encryption and decryption	* The protocol can withstand forgery, replay, and tag-tracking attacks.	* Does not provide any formal security analysis using AVISPA, Scyther tools, etc.
Ref [26]	* Lightweight	* Simple XOR * Improved permutation * Left rotation	* The scheme resists replay and de-synchronization attacks.	* Insecure against impersonation, secret disclosure, replay, traceability, man-in-the-middle, and de-synchronization attacks.
Ref [28]	* Lightweight	* Left rotation * Cross operation * PRNG * Bitwise XOR * Concatenation	* The protocol resists typical attacks including replay, Denial-of-Service (DoS), and synchronization attacks.	* Insecure against traceability, reader impersonation, and full and partial secret disclosure attacks.
Ref [33]	* Lightweight	* Vector-space * Basis of a vector-space * Linear Mapping * PRNG * Concatenation * Hash function	* The protocol can also withstand disclosure, replay, and tag location tracking attacks as well as ensures tag anonymity, mutual authentication, forward secrecy, etc.	* Does not provide any formal security analysis using AVISPA, Scyther tools, etc.
Ref [31]	* Ultralightweight	* Bitwise XOR * Reformation operation * Circular left–right rotations	* The scheme achieves tag location privacy, resists impersonation, disclosure, replay, and de-synchronization attacks.	* Insecure against insider and impersonation attacks.
Our	* Ultralightweight	* Bitwise XOR * PRNG * Circular left–right rotations * $Half\text{-}flip$ operation	* The scheme provides a stronger level of security and enhances the security functionalities.	* Not known yet

Table 2
Symbol and definitions.

Symbol	Description
$\mathcal{T}$	RFID Tag
$\mathcal{R}$	Reader
CS	Cloud server
$\mathcal{A}$	Adversary
k_i	Secret key shared between $\mathcal{T}$ and CS
K	Private key of CS
$rank(\mathcal{X})$	Number of 1's present in string $\mathcal{X}$
$nullity(\mathcal{Y})$	Number of 0's present in string $\mathcal{Y}$
$f_h(\mathcal{X},\mathcal{Y})$	$Half\text{-}flip$ operation between strings $\mathcal{X}$ and $\mathcal{Y}$
$Rot_l(\mathcal{X},\mathcal{Y})$	Circular left rotation of $\mathcal{X}$ by $rank(\mathcal{Y})$
$Rot_r(\mathcal{X},\mathcal{Y})$	Circular right rotation of $\mathcal{X}$ by $rank(\mathcal{Y})$
IDS	Index-pseudonym stored in $\mathcal{T}$ and database of CS
TID	Tag's identification number
ID	Static identification number of each RFID tag $\mathcal{T}$
n_1, n_2	Pseudo-random numbers generated at CS
$\oplus$	XOR operator
$\stackrel{?}{=}$	Comparison operator
N	Total number of tags stored in RFID systems
L	Length of each parameter (in bits)

3.3. Definition of half-flip

Suppose $\mathcal{X}$ and $\mathcal{Y}$ are two n-bit length strings, where

$$\mathcal{X} = x_n x_{n-1} \dots x_2 x_1, x_i \in \{0,1\}, i = 1,2,\dots,n,$$
$$\mathcal{Y} = y_n y_{n-1} \dots y_2 y_1, y_j \in \{0,1\}, j = 1,2,\dots,n.$$

The $half\text{-}flip$ of $\mathcal{X}$ with $\mathcal{Y}$ can be denoted as $f_h(\mathcal{X},\mathcal{Y})$. Therefore, the outcome can be written as

$$f_h(\mathcal{X},\mathcal{Y}) = \mathcal{Z} = z_n z_{n-1} \dots z_2 z_1$$
$$= Rot_r(\mathcal{X}'') \oplus Rot_l(\mathcal{Y}''), \quad z_k \in \{0,1\}, k = 1,2,\dots,n.$$

The newly proposed ultralightweight $half\text{-}flip$ can be divided into four steps such as rank and nullity, circular rotations, half-flipping, and XORing. To better understand the ultralightweight $f_h(\mathcal{X},\mathcal{Y})$ operation, we consider $\mathcal{X}$ and $\mathcal{Y}$ two 24-bits length strings as given below:

$\mathcal{X} = 101010111100010101111101$ and

$\mathcal{Y} = 110110111010011001010011.$

Step 1: **Rank and nullity** Compute rank and nullity of strings $\mathcal{X}$ and $\mathcal{Y}$ for 1's and 0's appear in $\mathcal{X}/\mathcal{Y}$

$rank(\mathcal{X})$ = Number of 1's presents in $\mathcal{X} = 15$ and

$\mathcal{X} = \sim(100010101111)$	101101010111
$\mathcal{X}' = 011101010000010010101000$	
$\mathcal{Y} = 100110010100$	$\sim(111101101110)$
$\mathcal{Y}' = 011001101011000010010001$	

Fig. 1. Computation of $\mathcal{X}'$ and $\mathcal{Y}'$.

$nullity(\mathcal{X})$ = Number of 0's presents in $\mathcal{X} = 9$

$rank(\mathcal{Y})$ = Number of 1's presents in $\mathcal{Y} = 14$ and

$nullity(\mathcal{Y})$ = Number of 0's presents in $\mathcal{Y} = 10$.

***Step 2:* Rotation** Perform circular left rotation on $\mathcal{X}$ by $nullity(\mathcal{X})$ and circular right rotation on $\mathcal{Y}$ by $rank(\mathcal{Y})$.

$Rot_l(\mathcal{X}, nullity(\mathcal{X})) = 100010101111101101010111$

$Rot_r(\mathcal{Y}, rank(\mathcal{Y})) = 100110010100111101101110$.

***Step 3:* Half flipping** Divide the strings $\mathcal{X}$ and $\mathcal{Y}$ into two equal parts. Then, as shown in Fig. 1 a simple $not(\sim)$ operation is performed on the first half of string $\mathcal{X}$ and the second half of string $\mathcal{Y}$ to compute $\mathcal{X}'$ and $\mathcal{Y}'$, respectively. Thereafter, the two equal substrings of $\mathcal{X}$ (i.e., $\mathcal{X}_1, \mathcal{X}_2$) and $\mathcal{Y}$ (i.e., $\mathcal{Y}_1, \mathcal{Y}_2$) is obtained, and perform subsequently the circular right rotation on $\mathcal{X}''$ by $rank(\mathcal{X})$ and the circular left rotation on $\mathcal{Y}''$ by $nullity(\mathcal{Y})$ as shown in Fig. 2.

***Step 4:* XORing** To compute the output of $half\text{-}flip$ operation, the bitwise XOR operation is performed between $Rot_r(\mathcal{X}'')$ and $Rot_l(\mathcal{Y}'')$. Then, we have

$$f_h(\mathcal{X}, \mathcal{Y}) = Rot_r(\mathcal{X}'', rank(\mathcal{X})) \oplus Rot_l(\mathcal{Y}'', nullity(\mathcal{Y})) = 000111010010011111000000$$

$rank(f_h(\mathcal{X}, \mathcal{Y})) = 9$ and $nullity(f_h(\mathcal{X}, \mathcal{Y})) = 15$.

4. System model and adversarial model

4.1. RFID system model

We present our formal privacy model in this section which is the same as the Vaudenay Privacy Model (VPM) [34,35] that ensures the privacy guarantees of our proposed scheme. This model is considered a powerful model among all existing methods. We provide the formal proof of two privacy notions, i.e., untraceability and information privacy, based on this model, these are discussed in Section 6.

The following components are comprised of an RFID scheme given below:

- **SetupServer** $(1^s) \rightarrow (IDS, DB)$
  This is an algorithm used to match IDS with stored IDS or IDS_{old}. If it finds a match, then for each tag, indexes IDS or IDS_{old} are used to fetch TID and k_i from the database. It also generates a database (DB), where all the secret information of the tags is stored.
- **SetupReader** $(1^s) \rightarrow (k_i)$
  Based on the security parameter s, a shared secret key k_i is generated by it, and it is shared with all the legitimate tags that are stored in the database of the cloud server.
- **SetupTag** $(TID) \rightarrow (k_i, S)$
  This is an algorithm used to generate a tag that has a secret key k_i, a tag's unique identification number TID, and an index-pseudonym IDS. It also generates an initial state S of the tag. The pair (TID, k_i) is stored for each legitimate tag in the database of a cloud server. Also, k_i is shared with all the legitimate readers.
- **IdenTag** $(\pi) \rightarrow Output$
  This is an interactive protocol π between RFID system components that is a tag $\mathcal{T}$, a reader $\mathcal{R}$, and the cloud server CS. If the server authenticates the tag, then $Output = TID$ (TID is legitimate), otherwise $Output = \perp$ (TID is not legitimate).

4.2. Adversarial model

This section discusses the capabilities of the adversaries. Our adversary model can capture a powerful adversary, where he/she can record or monitor all communications over the channel [36–38]. Moreover, an adversary $\mathcal{A}$ can eavesdrop or listen on the channel and obtain secret information in an RFID system. This model considers a tag as a free tag or drawn tag according to the accessibility of the tag to $\mathcal{A}$. The tag is considered a drawn tag if it is inside the read range of an adversary $\mathcal{A}$, otherwise, it is a free tag. The abilities of $\mathcal{A}$ can be represented by the following eight oracles as given below:

- **Create** $Tag^b(TID)$
  The adversary $\mathcal{A}$ creates a free tag with a tag's unique identifier (or unique identification number) TID. The free tag or drawn tag will be legitimate if $b = 1$, otherwise $b = 0$.

$$\texttt{DrawTag}(distr, n) \longrightarrow (vtag_1, b_1, vtag_2, b_2, \ldots, vtag_n, b_n)$$

The adversary $\mathcal{A}$ can access the randomly chosen set of tags from the set of free tags with a probability distribution $distr$. The free tag or drawn tag $vtag_i$ (virtual tag) will be legitimate if $b_i = 1$, otherwise $b_i = 0$.

- **Free** $(vtag)$
  Adversary $\mathcal{A}$ changes the status of drawn tag $vtag$ to free tag. So it makes $vtag$ untraceable. Hence, it cannot be traced by $\mathcal{A}$.
- **Launch** (π)
  The adversary $\mathcal{A}$ can launch a new protocol session run π.
- **SendReader** $(m, \pi) \rightarrow m'$
  This oracle sends a message m to a reader in a protocol session run π and receives the answer m' from the reader.
- **SendTag** $(m, vtag) \rightarrow m'$
  This oracle sends a message m to a virtual tag $vtag$ and receives the answer m' from the tag.
- **Result** (π)
  After completion of a new protocol session run π, it generates the output. If $\mathcal{A}$ succeeds then the $Output = 1$, otherwise $Output = \perp$.
- **Corrupt** $(vtag)$
  It returns the tag's initial state S. When a tag is not used for a long time after this oracle call, it means that the tag is destroyed.

5. Anonymous and reliable authentication scheme

The proposed scheme contains two phases namely setup and authentication phases.

5.1. AnonR^2AS: Assumptions

The following are noteworthy capabilities of an adversary as listed below:

1. The adversary can insert, delete modify, inject, and/or even listen to all the messages during communication.
2. The channel is secure between $\mathcal{R}$ and CS, while it is insecure between $\mathcal{T}$ and $\mathcal{R}$.
3. The CS is always considered to be an untrusted entity where an adversary can easily access the information of the tags.

5.2. AnonR^2AS: Setup phase

The three components (i.e., $\mathcal{T}$, $\mathcal{R}$, and CS) used in our proposed scheme store some information during the setup phase as shown in Fig. 3.

1. Initially, a unique static tag's identification number TID, a tag's private key k_i, and a index-pseudonym IDS are stored in each tag's internal memory.

$\mathcal{X}_1 = 011101010000$	$\mathcal{X}_2 = 010010101000$
$\mathcal{X}'' = \mathcal{X}_1 \| \mathcal{Y}_1 = 011101010000011001101011$	$rank(\mathcal{X}) = 15$
$Rot_r(\mathcal{X}'', rank(\mathcal{X})) = 000011001101011011101010$	
$\mathcal{Y}_1 = 011001101011$	$\mathcal{Y}_2 = 000010010001$
$Y'' = \mathcal{X}_2 \| \mathcal{Y}_2 = 010010101000000010010001$	$nullity(\mathcal{Y}) = 10$
$Rot_l(\mathcal{Y}'', nullity(\mathcal{Y})) = 000000100100010100101010$	

Fig. 2. Computation of $f_h(\mathcal{X}, \mathcal{Y})$.

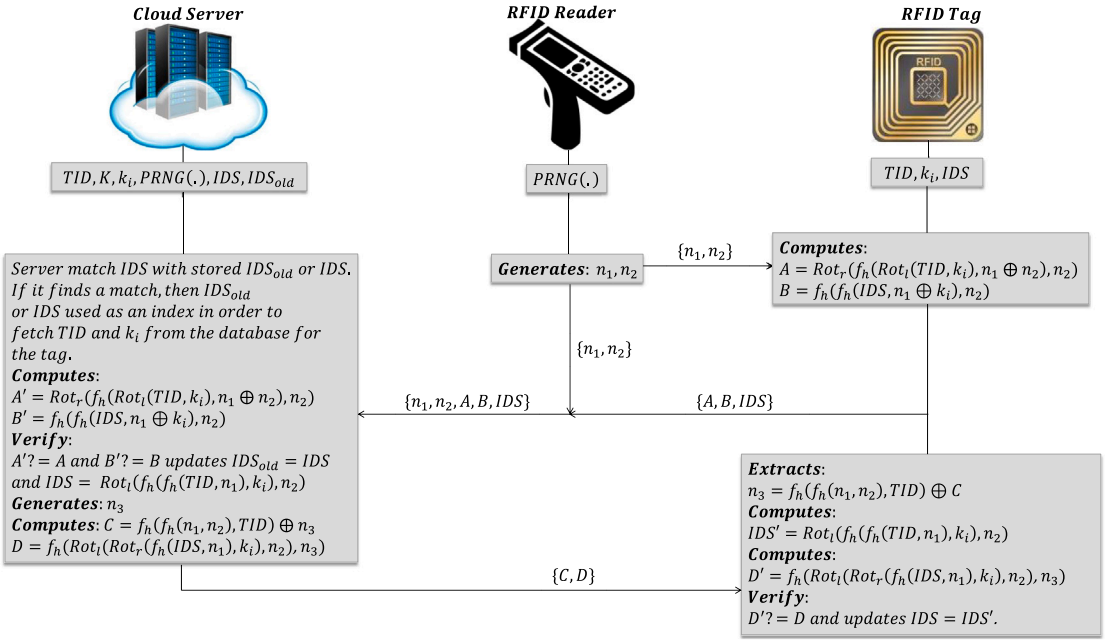


Fig. 3. Anonymous and reliable proposed authentication scheme.

2. For each tag, the tag's private parameter k_i and TID are stored with the cloud server in encrypted form using cloud servers' private key K.
3. The new and old pseudonym IDS, IDS_{old} are stored at cloud server in the form of plaintext. Initially, $IDS_{old} = IDS$ (as a random number) and $IDS = Null$.
4. The cloud server and each reader have a $PRNG(\cdot)$.

5.3. AnonR^{2}AS: Authentication phase

The execution steps of our AnonR^{2}AS scheme as shown in Fig. 3.

Step 1: $Msg_1 : \mathcal{R} \rightarrow \mathcal{T} : \{n_1, n_2\}$.

Initially, $\mathcal{R}$ generates two L-bit pseudo-random numbers n_1, n_2 and sends them to $\mathcal{T}$.

Step 2: $Msg_2 : \mathcal{T} \rightarrow \mathcal{R} : \{A, B, IDS\}$.

After receiving n_1 and n_2, $\mathcal{T}$ computes the responses A and B, and sends A and B along with IDS to $\mathcal{R}$.

- Computes $A = Rot_r(f_h(Rot_l(TID, k_i), n_1 \oplus n_2), n_2)$.
- Computes $B = f_h(f_h(IDS, n_1 \oplus k_i), n_2)$.

Step 3: $Msg_3 : \mathcal{R} \rightarrow CS : \{n_1, n_2, A, B, IDS\}$.

Upon receiving A, B, and IDS, $\mathcal{R}$ sends A, B and IDS along with n_1, n_2 to CS over a secure communication channel.

Step 4: $Msg_4 : CS \rightarrow \mathcal{T} : \{C, D\}$.

After receiving n_1, n_2, A, B, and IDS, CS matches IDS with stored IDS_{old} or IDS in the database. If it finds a match, then CS uses the IDS as an index to decrypt and fetch TID and k_i from the database for the tag using the private key K of the cloud server. The cloud server computes the local values of A', B', and checks whether $A' \stackrel{?}{=} A$ and $B' \stackrel{?}{=} B$, if so, CS authenticates $\mathcal{T}$ as a legitimate tag and updates IDS and IDS_{old}. Thereafter, $\mathcal{R}$ generates a L-bit pseudo-random number n_3 and subsequently computes the responses C and D, and further sends them to the tag via reader.

- Computes $A' = Rot_r(f_h(Rot_l(TID, k_i), n_1 \oplus n_2), n_2)$.
- Computes $B' = f_h(f_h(IDS, n_1 \oplus k_i), n_2)$.
- Verify $A' \stackrel{?}{=} A$ and $B' \stackrel{?}{=} B$, and updates $IDS_{old} = IDS$ and $IDS = Rot_l(f_h(f_h(TID, n_1), k_i), n_2)$.
- Computes $C = f_h(f_h(n_1, n_2), TID) \oplus n_3$.
- Computes $D = f_h(Rot_l(Rot_r(f_h(IDS', n_1), k_i), n_2), n_3)$.

Step 5: $Verification\ at\ Tag$.

Upon receiving C and D, $\mathcal{T}$ extracts the random number n_3 from C. The tag computes IDS' and local value of D' and checks whether $D' \stackrel{?}{=} D$, if so, $\mathcal{T}$ authenticates CS and updates its index-pseudonym IDS.

- Extracts $n_3 = f_h(f_h(n_1, n_2), TID) \oplus C$.
- Computes $IDS' = Rot_l(f_h(f_h(TID, n_1), k_i), n_2)$.
- Computes $D' = f_h(Rot_l(Rot_r(f_h(f_h(IDS', n_1), k_i), n_2), n_3)$.
- Verify $D' \stackrel{?}{=} D$, and updates $IDS = IDS'$.

6. Adversary classes

$\mathcal{A}$ is a probabilistic polynomial time ($p.p.t$) adversary that can arbitrarily access all oracles mentioned in Section 3. Depending upon

the Corrupt($vtag$) oracle, the adversaries can be classified into five classes according to VPM as Weak, Forward, Destructive, Strong, and Narrow are as follows:

- **Strong Adversary** ($\mathcal{A}_S$)
  This class of adversaries may use all the above eight oracles.
- **Destructive Adversary** ($\mathcal{A}_D$)
  This class of adversaries always destroys the corrupted tags. Besides, they do not use $vtag$ again after a Corrupt($vtag$) oracle.
- **Forward Adversary** ($\mathcal{A}_F$)
  This class of adversaries can only access the Corrupt($vtag$) oracle after the first call of Corrupt($vtag$).
- **Weak Adversary** ($\mathcal{A}_W$)
  This class of adversaries cannot access the Corrupt($vtag$) oracle.
- **Narrow Adversary** ($\mathcal{A}_N$)
  This class of adversaries cannot access Result (π) oracle.

Proposition 1. *AnonR²AS preserves the property of information privacy for the $\mathcal{A}_S$.*

Proof. Consider $\mathcal{A}$ executes the following steps for the information privacy experiment as illustrated in Algorithm 1.

- In an RFID system, $\mathcal{A}$ gets access to n-tags by invoking a DrawTag() oracle.

  $\text{DrawTag}(distr, n) \longrightarrow (vtag_1, b_1, vtag_2, b_2, \ldots, vtag_n, b_n)$

- $\mathcal{A}$ randomly selects one virtual tag $vtag_i$ from the above-drawn tag list and performs the following oracles as shown below:

Algorithm 1 Attack Game $\mathcal{AG}^{IPriv}$ for Information Privacy

1: **Experiment**: $Exp_{\mathcal{A}}^{IPriv}$
2: CreateTag(TID_i) and CreateTag(TID_j)
3: $vtag_b \longleftarrow$ DrawTag($TID)_b$
4: Launch(π)
5: $(n_1, n_2) \longleftarrow$ SendReader($\pi, init$)
6: $(A, B, IDS) \longleftarrow$ SendTag($vtag_b, n_1, n_2$)

To breach the information privacy, the adversary $\mathcal{A}$ needs to obtain the secrets used in A, B, C, and D. Fetching the secrets (i.e., k_i and TID) for an adversary $\mathcal{A}$ used in the above responses without physically tampered the virtual tag $vtag_i$ is infeasible. □

Proposition 2. *AnonR²AS preserves the property of untraceability for the $\mathcal{A}_S$.*

Proof. Consider $\mathcal{A}$ executes the following steps for the untraceability experiment as illustrated in 2.

- $\mathcal{A}$ gets access to n-tags by invoking a DrawTag() oracle.

  $\text{DrawTag}(distr, n) \longrightarrow (vtag_1, b_1, vtag_2, b_2, \ldots, vtag_n, b_n)$

- $\mathcal{A}$ randomly selects two uncorrupted virtual tags $vtag_i$ and $vtag_j$ corresponding to $vtag_b$, where $b \in \{i, j\}$. The adversary starts an analysis of $vtag_b$ using the following oracles as shown below:

The above experiment shows that the adversary $\mathcal{A}$ has succeeded in his/her experiment. $\mathcal{A}$ can obtain whether the response message(s), i.e., A, B, C, or D, are produced by $vtag_i$ or $vtag_j$. In each authentication session, our AnonR²AS scheme utilizes random numbers in each response message. So, it is hard to guess the correct value of the actual tag by an adversary. Hence, the untraceability property is preserved in our proposed scheme. □

Algorithm 2 Attack Game $\mathcal{AG}^{UNT}$ for RFID Location Privacy: Untraceability

1: **Experiment**: $Exp_{\mathcal{A}}^{UNT}$
2: CreateTag(TID_i) and CreateTag(TID_j)
3: $vtag_b \longleftarrow$ DrawTag(TID_b)
4: Launch(π)
5: $(n_1', n_2') \longleftarrow$ SendReader($\pi, init$)
6: $(A', B', IDS') \longleftarrow$ SendTag($vtag_b, n_1', n_2'$)
7: FreeTag($vtag_i$)
8: $\mathcal{A}$ terminates execution of the experiment and produces outputs a guess bit b'
9: **if** $\mathcal{A}$ succeeds **then**
10: $1 \longleftarrow b'$
11: **else**
12: $0 \longleftarrow b'$
13: **end if**

Table 3
Comparison of security and privacy features.

Scheme ↓ ⟶	Ref [21]	Ref [22]	Ref [24]	Ref [26]	Our
SPF_1	✘	✔	❀	✔	✔
SPF_2	✘	✔	✔	✔	✔
SPF_3	✘	✘	✘	✘	✔
SPF_4	✘	✘	✔	✔	✔
SPF_5	✘	✘	✔	✔	✔
SPF_6	✔	✔	✔	✘	✔
SPF_7	✔	✔	✔	✔	✔
SPF_8	✔	✔	✔	✘	✔

Acronyms SPF_1: Mutual authentication; SPF_2: Confidentiality; SPF_3: Tag anonymity; SPF_4: Tag location privacy; SPF_5: Secure against de-synchronization attack; SPF_6: Secure against impersonation attack; SPF_7: Secure against replay attack; SPF_8: Secure against disclosure attack; ✔: Satisfied; ✘: Not satisfied; ❀: Partially satisfied.

7. Analysis and evaluation

This section discusses the security and privacy as well as performance analysis of the AnonR²AS scheme and provides a comparison with the state-of-the-art schemes.

7.1. Informal security analysis

The proposed AnonR²AS scheme performs the security analysis comparison with four different RFID authentication schemes (such as Ref. [21], Ref. [22], Ref. [24], and Ref. [26]) in terms of various security and privacy features. The security of any RFID authentication scheme(s) is considered one of the most important aspects. Compared to previous existing schemes, Table 3 demonstrates that our scheme satisfies all necessary security features.

- **Mutual authentication**
  The cloud server authenticates tag and vice-versa. In our AnonR²AS scheme, the tag is authenticated by the cloud server by comparing the computed response messages $A' \stackrel{?}{=} A$ and $B' \stackrel{?}{=} B$. On the other hand, cloud server is authenticated by the tag comparing $D' \stackrel{?}{=} D$. The response messages C and D are computed by using stored shared secrets (TID, k_i, and IDS) on the tag. The server matches IDS with stored IDS and IDS_{old}. If it is successful, the IDS and IDS_{old} are used to fetch TID and k_i for the tag from the encrypted cloud database using its private key K. In addition, the cloud generates a pseudo-random number n_3 to compute C and D. Therefore, mutual authentication is accomplished between legal participants in our proposed scheme.
- **Tag anonymity**
  We adopt the concept of index-pseudonym IDS to protect the anonymity of tags. However, IDS is an index of the table in

which all the secrets of the tags are stored. The tag does not reveal its identity TID and uses IDS as identity. In particular, the IDS is updated after each successful protocol authentication session and the update operations involve random numbers and the private key k_i of the tag $\mathcal{T}$. So, the tag will be anonymous and cannot be identified or tracked by the adversary. Moreover, the proposed scheme does not use any unbalanced operations in the updating process. Therefore, there is no provision for tag tracking through IDS. All the messages also contain random numbers and thus the tags cannot be tracked.

- **Replay attack**
  $\mathcal{A}$ transmits the previous tampered messages to the cloud server to prove that he/she is a legitimate entity. $\mathcal{A}$ can intercept or eavesdrop on the communicated messages over the channel in an authentication session. The adversary may collect and eavesdrop on data between tag and reader, reuse this data, and send it repeatedly. over an insecure wireless channel. Consider an adversary who may collect and eavesdrop on the tag's responses A and B and replay the previously captured messages A and B towards the reader. In this case, the reader cannot authenticate such messages, because $\mathcal{R}$ generates two fresh pseudo-random numbers n_1 and n_2 in each authentication session. The reader transmits the messages A, B, IDS along with two different pseudo-random numbers n_1' and n_2'. But $\mathcal{R}$ stores two fresh random numbers n_1 and n_2, so $A' \neq A$ and $B' \neq B$. Hence, CS does not authenticate $\mathcal{T}$.
- **Impersonation attack**
  Consider $\mathcal{A}$ can eavesdrop on an authentication session and collect data from the responses A, B, C, and D. $\mathcal{A}$ must be able to compute the legitimate responses (such as A, B, C, and D) to impersonate the tag. However, it is hard for an adversary to compute such responses without knowing TID, K, k_i, and session-specific random numbers n_1, n_2. Thus, our AnonR^{2}AS scheme resists the impersonation attack.
- **De-synchronization attack**
  In our AnonR^{2}AS scheme, the shared secrets cannot be de-synchronized between $\mathcal{T}$ and $\mathcal{R}$ by an adversary. To update the shared secret, $\mathcal{T}$ and $\mathcal{R}$ use different random numbers in each authentication phase. The cloud server keeps two identities IDS and IDS_{old} to verify the identity of a tag. Even if adversary $\mathcal{A}$ able to stop the last message containing C, D with hidden information of IDS, the tag can use IDS_{old} to authenticate itself from the cloud server. Moreover, $\mathcal{A}$ cannot modify the responses A, B, C, and D. Hence, our AnonR^{2}AS scheme successfully resists the de-synchronization attack.
- **Tag location privacy**
  The real identity TID of the tag is never exposed on the open channel rather an index-pseudonym IDS is used for authentication tasks and this IDS is updated after each successful authentication session. Therefore, using IDS an attacker can never identify whether two different sessions are initiated by the same tag or not. Hence, our scheme archives tag location privacy.
- **Disclosure attack**
  $\mathcal{A}$ can perform slight modifications on the response messages and send them to the tag to verify the modification's correctness. In our AnonR^{2}AS scheme, the simple XOR operation cannot disclose secrets in an authentication session. However, the adversary cannot determine any secrets from the computed response messages. Thus, our AnonR^{2}AS scheme is secure under the disclosure attack.

7.2. Performance evaluation

The performance evaluation of our AnonR^{2}AS is provided in Tables 4, 5, 6, and 7. The memory and computational capabilities of the tags are excessively limited when compared to $\mathcal{R}$ and CS. Therefore, we do only performance analysis for tags. A comparison has been carried out of our proposed AnonR^{2}AS scheme with existing schemes named Ref. [21], Ref. [22], Ref. [24], and Ref. [26]. As compared to these existing schemes, Tables 5 and 6 show that the communication costs (from $\mathcal{T} \longrightarrow \mathcal{R}$ and $\mathcal{R} \longrightarrow \mathcal{T}$) and storage costs (on $\mathcal{T}$ and $\mathcal{R}$+CS) of our scheme are relatively less and satisfactory. Also, our scheme guarantees a higher level of security with low computational complexity of the tags.

Table 4
Comparison of computational complexity.

Scheme ↓ ⟶	Entity	Ref [21]	Ref [22]	Ref [24]	Ref [26]	Our
Hash functions	$\mathcal{T}$	4	5	6	0	0
	$\mathcal{R}+CS$	$4+N$	5	10	0	0
$Half\text{-}flip$ operations	$\mathcal{T}$	0	0	0	0	7
	$\mathcal{R}+CS$	0	0	0	0	9
$PRNG(\cdot)$	$\mathcal{T}$	1	1	1	0	0
	$\mathcal{R}+CS$	1	1	1	0	3
$E(\cdot)_k/D(\cdot)_k$ operations	$\mathcal{T}$	0	0	0	0	0
	$\mathcal{R}+CS$	3	2	3	4	0
Permutation operations	$\mathcal{T}$	0	0	0	2	0
	$\mathcal{R}+CS$	0	0	0	4	0
Rotation operations	$\mathcal{T}$	0	0	0	1	5
	$\mathcal{R}+CS$	0	0	0	1	5

Table 5
Comparison of communication cost and communication round.

Scheme ↓ ⟶	Entity	Communication cost	Communication round
Ref [21]	$\mathcal{T} \longrightarrow \mathcal{R}$	$3L = 3 \times 96 = 288$-bits	10
	$\mathcal{R} \longrightarrow \mathcal{T}$	$3L = 3 \times 96 = 288$-bits	
Ref [22]	$\mathcal{T} \longrightarrow \mathcal{R}$	$3L = 3 \times 96 = 288$-bits	8
	$\mathcal{R} \longrightarrow \mathcal{T}$	$2L = 2 \times 96 = 192$-bits	
Ref [24]	$\mathcal{T} \longrightarrow \mathcal{R}$	$3L = 3 \times 96 = 288$-bits	7
	$\mathcal{R} \longrightarrow \mathcal{T}$	$2L = 2 \times 96 = 192$-bits	
Ref [26]	$\mathcal{T} \longrightarrow \mathcal{R}$	$4L = 4 \times 96 = 384$-bits	7
	$\mathcal{R} \longrightarrow \mathcal{T}$	$3L = 3 \times 96 = 288$-bits	
Our	$\mathcal{T} \longrightarrow \mathcal{R}$	$3L = 3 \times 96 = 288$-bits	4
	$\mathcal{R} \longrightarrow \mathcal{T}$	$2L = 2 \times 96 = 192$-bits	

Table 6
Comparison of storage cost.

Scheme ↓⟶	Entity	Storage cost
Ref [21]	$\mathcal{T}$	$3L = 3 \times 96 = 288$-bits
	$\mathcal{R}+CS$	$7NL = 7 \times 100 \times 96 = 67200$-bits
Ref [22]	$\mathcal{T}$	$2L = 2 \times 96 = 192$-bits
	$\mathcal{R}+CS$	$6NL = 6 \times 100 \times 96 = 57600$-bits
Ref [24]	$\mathcal{T}$	$3L = 3 \times 96 = 288$-bits
	$\mathcal{R}+CS$	$6NL = 6 \times 100 \times 96 = 57600$-bits
Ref [26]	$\mathcal{T}$	$5L = 5 \times 96 = 480$-bits
	$\mathcal{R}+CS$	$9NL = 9 \times 100 \times 96 = 86400$-bits
Our	$\mathcal{T}$	$3L = 3 \times 96 = 288$-bits
	$\mathcal{R}+CS$	$5NL = 5 \times 100 \times 96 = 48000$-bits

- **Computational complexity**
  It states the number of ultralightweight primitives used for each tag. There is no use of encryption $E(\cdot)_k$/decryption $D(\cdot)_k$ algorithm and one-way hash in our proposed AnonR^{2}AS scheme, such operations generally require high computational resources. We use only simple bitwise XOR ($\oplus$), circular left–right rotations $Rot_{l\ or\ r}(X, Y)$, and a $half\text{-}flip\ f_h(X, Y)$ operation as shown in Table 4. Since the running time of such operations is considered as negligible in comparison to one-way hash and encryption and decryption algorithms. Hence, we conclude that our AnonR^{2}AS scheme reduces the computational overheads on RFID tags.
- **Communication cost**

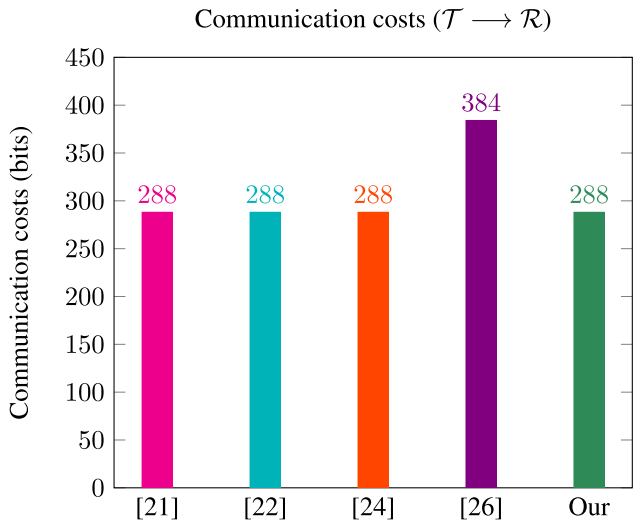


Fig. 4. Communication overhead from tag to reader.

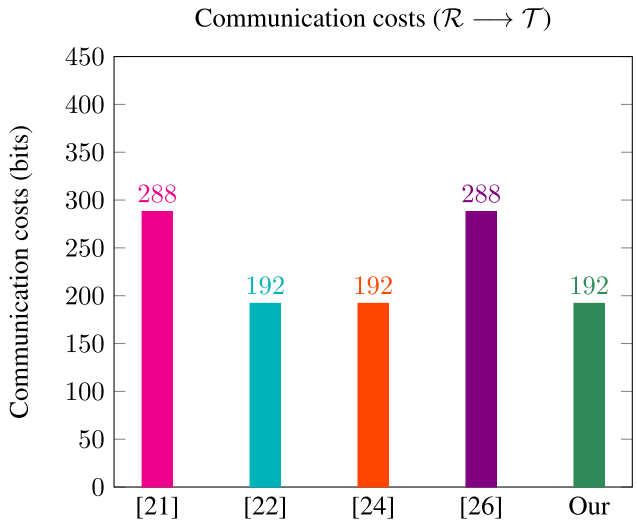


Fig. 5. Communication overhead from reader to tag.

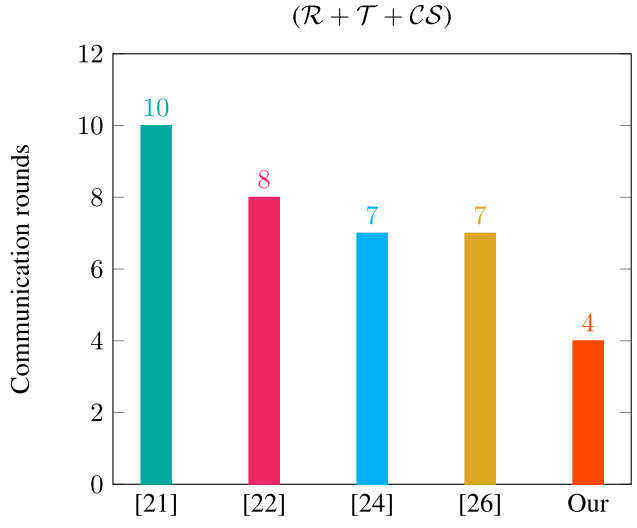


Fig. 6. Total communication rounds.

Table 7
Comparison of server search complexity (scalability).

Scheme ↓⟶	Ref [21]	Ref [22]	Ref [24]	Ref [26]	Our
Scalability	$\mathcal{O}(N)$	$\mathcal{O}(N)$	$\mathcal{O}(N)$	$\mathcal{O}(N)$	$\mathcal{O}(1)$

The total number of messages transmitted by the tag in each authentication session. In our AnonR^{2}AS scheme, there is a total of three communication messages that are transmitted by the tag. Thus, the communication cost consumes $3L$ bits on the tags. Further, we have assumed that the RFID EPC tag consumes a 96-bits for each parameter. Table 5 shows that our scheme consumes the total communication cost ($3L = 3 \times 96$) = 288-bits from ($\mathcal{T} \longrightarrow \mathcal{R}$), which is same as Ref. [21], Ref. [22], and Ref. [24]. Additionally, our scheme consumes a total communication cost ($2L = 2 \times 96$) = 192-bits from ($\mathcal{R} \longrightarrow \mathcal{T}$), which is same as Ref. [22] and Ref. [24]. Hence, our proposed AnonR^{2}AS scheme has a relatively low communication cost as compared to other existing schemes. Besides, a total of 4 communication rounds were consumed for mutual authentication, which is relatively less than Ref. [21], Ref. [22], Ref. [24], and Ref. [26], with 10, 8, 7, and 7 rounds, respectively. For better understanding, Figs. 4, 5, and 6 depict the graphical comparisons for communication costs and total communication rounds of our scheme against various state-of-the-art schemes.

- **Storage cost**
The number of key elements and the static tag ID are stored in the tag's internal memory space. In our AnonR^{2}AS scheme, the tag stores a total of three strings including the TID, an index-pseudonym IDS, and a secret key k_i. Therefore, the storage cost of each tag is $3L$-bits. Similarly, a tag population of up to $N = 100$ tags is considered by the database of an RFID system. Each parameter (or static identifier stored in ROM) consists of 96 bits (i.e., $L = 96$-bits) for each RFID EPC tag corresponding to EPCglobal. The proposed scheme consumes a total storage cost ($3L = 3 \times 96$) = 288-bits on $\mathcal{T}$, which is less than the cost of Ref. [26], but equal to the cost of Ref. [21] and Ref. [24], and slightly higher than the cost of Ref. [22]. However, consumes a total storage cost ($5NL = 5 \times 100 \times 96$) = 48,000-bits on ($\mathcal{R} + \mathcal{CS}$), which is less than the costs of Ref. [21], Ref. [22], Ref. [24], and Ref. [26]. Table 6 shows that our proposed AnonR^{2}AS scheme has a relatively low storage cost on $\mathcal{T}$ and ($\mathcal{R} + \mathcal{CS}$) as compared to other existing schemes. The storage costs of our scheme are compared with various state-of-the-art schemes as shown in Figs. 8 and 9.
- **Server search complexity (scalability)**
The overall analysis shows that our proposed AnonR^{2}AS scheme is preferable for low-cost RFID systems with less computational complexity and memory space, and it shows better performance in IoT environment. Moreover, the scalability property of an RFID authentication scheme can be computed by the cloud server to search for a match record in its database. The cloud server performs an exhaustive search operation for identifying an RFID tag in the existing schemes, hence these schemes cannot withstand the scalable property. In a single search attempt, Table 7 shows that the cloud server finds the matched records (i.e., k_i, IDS, and IDS_{old}) from the database. Thus, the server search complexity of our AnonR^{2}AS scheme is only $\mathcal{O}(1)$, which is highly scalable when comparing other existing schemes.

7.3. Formal verification using Scyther tool

This section presents a formal security verification which is done by using a robust and widely accepted verification tool named Scyther [39]. To simulate our proposed protocol, we first installed GraphViz library,

Scyther results : verify

Claim				Status	Comments
MyProposed	Tag	MyProposed,Tag1	Secret TID	Ok	No attacks within bounc
		MyProposed,Tag2	Secret ki	Ok	No attacks within bounc
		MyProposed,Tag3	Secret n1	Ok	No attacks within bounc
		MyProposed,Tag4	Secret n2	Ok	No attacks within bounc
		MyProposed,Tag5	Niagree	Ok	No attacks within bounc
		MyProposed,Tag6	Nisynch	Ok	No attacks within bounc
		MyProposed,Tag7	Alive	Ok	No attacks within bounc
		MyProposed,Tag8	Weakagree	Ok	No attacks within bounc
	Reader	MyProposed,Reader1	Secret ki	Ok	No attacks within bounc
		MyProposed,Reader2	Niagree	Ok	No attacks within bounc
		MyProposed,Reader3	Nisynch	Ok	No attacks within bounc
		MyProposed,Reader4	Alive	Ok	No attacks within bounc
		MyProposed,Reader5	Weakagree	Ok	No attacks within bounc
	CloudServer	MyProposed,CloudServer1	Secret TID	Ok	No attacks within bounc
		MyProposed,CloudServer2	Secret ki	Ok	No attacks within bounc
		MyProposed,CloudServer3	Secret n1	Ok	No attacks within bounc
		MyProposed,CloudServer4	Secret n2	Ok	No attacks within bounc
		MyProposed,CloudServer5	Niagree	Ok	No attacks within bounc
		MyProposed,CloudServer6	Nisynch	Ok	No attacks within bounc
		MyProposed,CloudServer7	Alive	Ok	No attacks within bounc
Done.		MyProposed,CloudServer8	Weakagree	Ok	No attacks within bounc

Fig. 7. Scyther verification results.

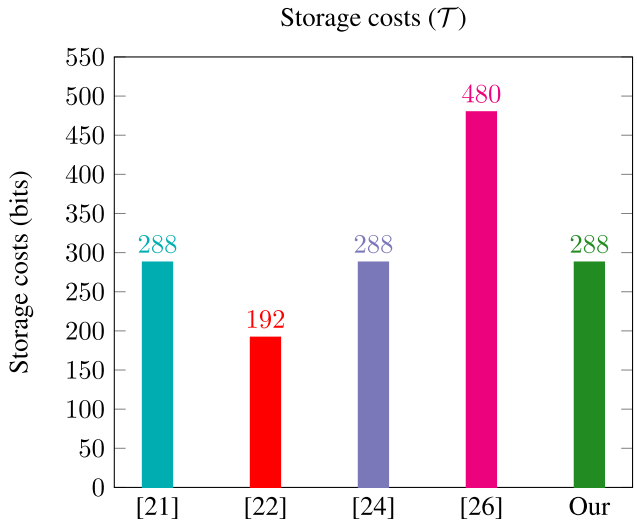


Fig. 8. Storage costs on tag.

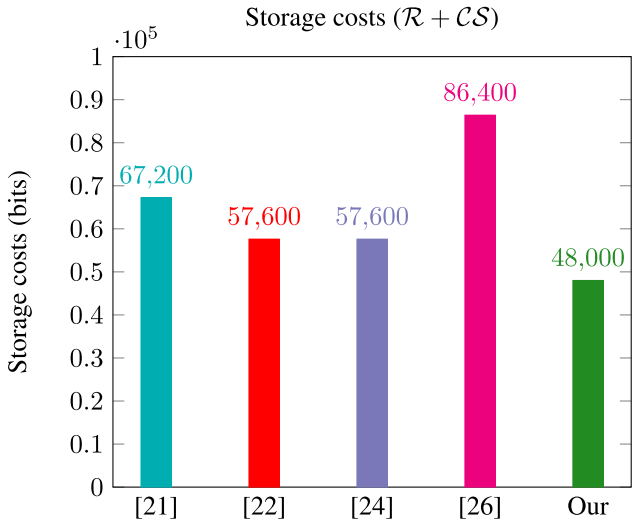


Fig. 9. Storage costs on reader and cloud server.

```
usertype TimeStamp,Data,Key,Nonce,Data;
const XOR: Function;
const LeftRot: Function;
const RightRot: Function;
const Halfflip: Function;
protocol MyProposed(Tag,Reader,CloudServer)
{
role Tag{
const A,A',B,B',C,C',D,D',ki,IDS,IDS',TID,IDSnew,n3;
var n1,n2: Nonce;
recv_!1(Reader,Tag,n1,n2);
macro A=RightRot(Halfflip(LeftRot(TID,ki),XOR(n1,n2)),n2);
macro B=Halfflip(Halfflip(IDS,XOR(n1,ki)),n2);
send_!2(Tag,Reader,A,B,IDS);
recv_!4(CloudServer,Tag,C,D);
macro n3=XOR(Halfflip(Halfflip(n1,n2),TID),C);
macro IDS'=LeftRot(Halfflip(Halfflip(TID,n1),ki),n2);
macro D'=Halfflip(LeftRot(RightRot(Halfflip(IDSnew,n1),ki),n2),n3);
match(D,D');
macro IDS=IDS';
claim(Tag, Secret, TID);
claim(Tag, Secret, ki);
claim(Tag, Secret, n1);
claim(Tag, Secret, n2);
claim(Tag, Niagree);
claim(Tag, Nisynch);
claim(Tag, Alive);
claim(Tag, Weakagree);
}
```

Fig. 10. SPDL code of tag role.

```
role Reader{
const A,A',B,B',C,C',D,D',ki,IDS,IDS',TID,IDSnew;
fresh n1,n2: Nonce;
send_!1(Reader,Tag,n1,n2);
recv_!2(Tag,Reader,A,B,IDS);
send_!3(Reader,CloudServer,{n1,n2,A,B,IDS}pk(CloudServer));
claim(Reader, Secret, ki);
claim(Reader, Niagree);
claim(Reader, Nisynch);
claim(Reader, Alive);
claim(Reader, Weakagree);
}
```

Fig. 11. SPDL code of reader role.

Python 2.7, wxPython libraries, and Scyther tool packages. Thereafter, we performed the simulation on Ubuntu 23.04, Core (TM) i5 processor, 8 GB DDR4-3200 MHz RAM, and 512 GB SSD Hard Drive. The Scyther is a simulation tool used for security protocol verification based on perfect cryptographic assumptions. Under these assumptions, the adversary cannot learn anything from the encrypted messages without knowing the decryption key.

The Scyther tool finds the problems that arise during the protocol design. In the Scyther simulation, a variety of security protocols can be either proven correct or false. Moreover, based on the verification results and claims in the protocols, we can find whether the protocol is correct or whether attacks can be found. If the protocol claims are incorrect, then it shows that there exist some attacks on the protocol. The Security Protocol Description Language (SPDL) is a Scyther's input language used to write the specification of security protocols. The specifications define a sequence of roles namely the Tag role in Fig. 10, Reader role in Fig. 11, and Cloud-Server role in Fig. 12 for the proposed scheme. Each role consists of a sequence of events namely send,

```
role CloudServer{
const A,A',B,B',C,C',D,D',ki,IDS,IDS',TID,IDSnew;
var n1,n2: Nonce;
recv_!3(Reader,CloudServer,{n1,n2,A,B,IDS}pk(CloudServer));
macro A'=RightRot(Halfflip(LeftRot(TID,ki),XOR(n1,n2)),n2);
macro B'=Halfflip(Halfflip(IDS,XOR(n1,ki)),n2);
match(A,A');
match(B,B');
macro IDS'=LeftRot(Halfflip(Halfflip(TID,n1),ki),n2);
fresh n3: Nonce;
macro C=XOR(Halfflip(Halfflip(n1,n2),TID),n3);
macro D=Halfflip(LeftRot(RightRot(Halfflip(IDSnew,n1),ki),n2),n3);
send_!4(CloudServer,Tag,C,D);
claim(CloudServer, Secret, TID);
claim(CloudServer, Secret, ki);
claim(CloudServer, Secret, n1);
claim(CloudServer, Secret, n2);
claim(CloudServer, Niagree);
claim(CloudServer, Nisynch);
claim(CloudServer, Alive);
claim(CloudServer, Weakagree);
}
}
```

Fig. 12. SPDL code of cloud server role.

receive, declarations, and claim [39]. In SPDL, several predefined claim events (such as Secret, Niagree, Nisynch, Aliveness, Weak agreement, and *Session-key reveal*) are used in each role specification to ensure security properties [40]. The match function in SPDL performs a logical comparison to verify the integrity of the messages. We have claimed the parameters such as TID, k_i, n_1, and n_2 on the tag and the cloud server as well as k_i on the reader. Fig. 7 shows the obtained simulation results using the Scyther simulation tool which summarizes that no attack is found on each of the claims specified in our scheme. All aforementioned secret information related to our scheme is protected against the adversary in a protocol session run.

8. Conclusion

In the past decades, RFID has shown prosperous developments in the area of automatic identification and data capture. Recently, a variety of cloud-based RFID authentication schemes for low-cost RFID systems have been proposed. These schemes cannot achieve many security requirements and are insecure against various security attacks. To fix these security flaws, we proposed an anonymous and reliable ultralightweight RFID-enabled authentication scheme for IoT systems in a cloud computing environment named AnonR^{2}AS. Our AnonR^{2}AS scheme utilizes simple XOR, left and right rotations, and an ultralightweight $half\text{-}flip$ operation, providing a stronger security level and enhancing the security functionalities. The proposed AnonR^{2}AS scheme resists known security attacks. The formal security verification is also done by the Scyther simulation tool which shows that no attack is found in our scheme. In addition, the information privacy and tag untraceability properties have been preserved by using Vaudenay privacy model. Furthermore, the performance analysis demonstrates a low computational overhead of the tags and shows suitability for low-cost RFID systems. This research can be applied in real-world IoT applications which is more suitable for the IoT environment.

Our scheme has some limitations regarding security and privacy i.e., it is not secure against physical and cloning security attacks. On the other hand, the cloud server is considered too powerful, if it is compromised, then an adversary can easily know all the secret parameters which results in performing some forgery attacks. In the future, we aim to overcome the security issues mentioned above.

CRediT authorship contribution statement

Mohd Shariq: Writing – original draft, Methodology, Formal analysis. **Mauro Conti:** Writing – review & editing, Supervision, Investigation. **Karan Singh:** Supervision, Investigation, Conceptualization. **Chhagan Lal:** Writing – review & editing, Validation, Conceptualization. **Ashok Kumar Das:** Validation, Supervision, Formal analysis. **Shehzad Ashraf Chaudhry:** Writing – review & editing, Investigation, Conceptualization. **Mehedi Masud:** Writing – review & editing, Validation, Funding acquisition, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgment

The authors extend their appreciation to Taif University, Saudi Arabia, for supporting this work through project number (TU-DSPP-2024-04).

Funding statement

This research was funded by Taif University, Saudi Arabia, project number (TU-DSPP-2024-04).

References

[1] A. Barati, A. Movaghar, M. Sabaei, RDTP: Reliable data transport protocol in wireless sensor networks, Telecommun. Syst. 62 (2016) 611–623.
[2] A. Zaslavsky, C. Perera, D. Georgakopoulos, Sensing as a service and big data, 2013, arXiv preprint arXiv:1301.0159.
[3] W. Kassab, K.A. Darabkh, A-Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations, J. Netw. Comput. Appl. 163 (2020) 102663.
[4] M. Al-Emran, S.I. Malik, M.N. Al-Kabi, A survey of internet of things (IoT) in education: opportunities and challenges, in: Toward Social Internet of Things (SIoT): Enabling Technologies, Architectures and Applications, Springer, 2020, pp. 197–209.
[5] S.A. Chaudhry, M.S. Farash, N. Kumar, M.H. Alsharif, PFLUA-DIoT: A pairing free lightweight and unlinkable user access control scheme for distributed IoT environments, IEEE Syst. J. 16 (1) (2022) 309–316, http://dx.doi.org/10.1109/JSYST.2020.3036425.
[6] X. Jia, Q. Feng, C. Ma, An efficient anti-collision protocol for RFID tag identification, IEEE Commun. Lett. 14 (11) (2010) 1014–1016.
[7] X. Jia, Q. Feng, T. Fan, Q. Lei, RFID technology and its applications in Internet of Things (IoT), in: 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), IEEE, 2012, pp. 1282–1285.
[8] N. Pakniat, Z. Eslami, Cryptanalysis and improvement of a group RFID authentication protocol, Wirel. Netw. 26 (5) (2020) 3363–3372.
[9] Q.Z. Sheng, X. Li, S. Zeadally, Enabling next-generation RFID applications: Solutions and challenges, Computer 41 (9) (2008) 21–28.
[10] D. He, S. Zeadally, An analysis of RFID authentication schemes for internet of things in healthcare environment using elliptic curve cryptography, IEEE Int. Things J. 2 (1) (2014) 72–83.
[11] M. Shariq, K. Singh, A secure and lightweight RFID-enabled protocol for IoT healthcare environment: A vector space based approach, Wirel. Pers. Commun. 127 (4) (2022) 3467–3491.
[12] M. Ataei Nezhad, H. Barati, A. Barati, An authentication-based secure data aggregation method in internet of things, J. Grid Comput. 20 (3) (2022) 29.
[13] W. Liu, Research on cloud computing security problem and strategy, in: 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), IEEE, 2012, pp. 1216–1219.
[14] Z. Wei, Y. Zhang, J. Yang, Private assets protection system based on RFID and cloud computing, in: Proceedings of the International Conference on Electronics, Communications and Networks (CECNet), 2013, pp. 196–198.
[15] P. Alimoradi, A. Barati, H. Barati, A hierarchical key management and authentication method for wireless sensor networks, Int. J. Commun. Syst. 35 (6) (2022) e5076.
[16] S.A. Khah, A. Barati, H. Barati, A dynamic and multi-level key management method in wireless sensor networks (WSNs), Comput. Netw. 236 (2023) 109997.
[17] H. Shi, X. Bai, C. Ren, C. Zhao, Development of internet of vehicle's information system based on cloud, J. Softw. 9 (7) (2014) 1848–1853.
[18] S. Anandhi, R. Anitha, V. Sureshkumar, An authentication protocol to track an object with multiple RFID tags using cloud computing environment, Wirel. Pers. Commun. 113 (4) (2020) 2339–2361.
[19] M. Shariq, K. Singh, M.Y. Bajuri, A.A. Pantelous, A. Ahmadian, M. Salimi, A secure and reliable RFID authentication protocol using digital schnorr cryptosystem for IoT-enabled healthcare in COVID-19 scenario, Sustainable Cities Soc. 75 (2021) 103354.
[20] M. Shariq, K. Singh, P.K. Maurya, A. Ahmadian, D. Taniar, AnonSURP: an anonymous and secure ultralightweight RFID protocol for deployment in internet of vehicles systems, J. Supercomput. 78 (6) (2022) 8577–8602.
[21] W. Xie, L. Xie, C. Zhang, Q. Zhang, C. Tang, Cloud-based RFID authentication, in: 2013 IEEE International Conference on RFID, RFID, IEEE, 2013, pp. 168–175.
[22] S. Abughazalah, K. Markantonakis, K. Mayes, Secure improved cloud-based RFID authentication protocol, in: Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance, Springer, 2014, pp. 147–164.
[23] B. Surekha, K.L. Narayana, P. Jayaprakash, C.S. Vorugunti, A realistic lightweight authentication protocol for securing cloud based RFID system, in: 2016 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM, IEEE, 2016, pp. 54–60.
[24] H. Xiao, A.A. Alshehri, B. Christianson, A cloud-based RFID authentication protocol with insecure communication channels, in: 2016 IEEE Trustcom/BigDataSE/ISPA, IEEE, 2016, pp. 332–339.
[25] K. Fan, Q. Luo, H. Li, Y. Yang, Cloud-based lightweight RFID mutual authentication protocol, in: 2017 IEEE Second International Conference on Data Science in Cyberspace, DSC, IEEE, 2017, pp. 333–338.
[26] K. Fan, Q. Luo, K. Zhang, Y. Yang, Cloud-based lightweight secure RFID mutual authentication protocol in IoT, Inform. Sci. 527 (2020) 329–340.
[27] M. Adeli, N. Bagheri, S. Sadeghi, S. Kumari, χ Perbp: a cloud-based lightweight mutual authentication protocol, IACR Cryptol. ePrint Arch. 2021 (2021) 144.
[28] K. Fan, W. Jiang, H. Li, Y. Yang, Lightweight RFID protocol for medical privacy protection in IoT, IEEE Trans. Ind. Inform. 14 (4) (2018) 1656–1665.
[29] S.F. Aghili, H. Mala, P. Kaliyar, M. Conti, SecLAP: Secure and lightweight RFID authentication protocol for Medical IoT, Future Gener. Comput. Syst. 101 (2019) 621–634.
[30] M. Safkhani, Y. Bendavid, S. Rostampour, N. Bagheri, On designing lightweight RFID security protocols for medical IoT, IACR Cryptol. ePrint Arch. 2019 (2019) 851.
[31] A. Kumar, K. Singh, M. Shariq, C. Lal, M. Conti, R. Amin, S.A. Chaudhry, An efficient and reliable ultralightweight RFID authentication scheme for healthcare systems, Comput. Commun. 205 (2023) 147–157.
[32] I. Ebrahimi, M. Nikooghadam, An efficient secure channel establishment through lightweight key distribution in e-health communication systems, Multimedia Tools Appl. (2024) 1–21.
[33] M. Shariq, K. Singh, A novel vector-space-based lightweight privacy-preserving rfid authentication protocol for IoT environment, J. Supercomput. (2021) 1–31.
[34] S. Vaudenay, On privacy models for RFID, in: International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2007, pp. 68–87.
[35] M.R. Alagheband, M.R. Aref, Simulation-based traceability analysis of RFID authentication protocols, Wirel. Pers. Commun. 77 (2) (2014) 1019–1038.
[36] M.W. Akram, A.K. Bashir, S. Shamshad, M.A. Saleem, A.A. AlZubi, S.A. Chaudhry, B.A. Alzahrani, Y.B. Zikria, A secure and lightweight drones-access protocol for smart city surveillance, IEEE Trans. Intell. Transp. Syst. 23 (10) (2022) 19634–19643, http://dx.doi.org/10.1109/TITS.2021.3129913.
[37] S.A. Chaudhry, J. Nebhan, K. Yahya, F. Al-Turjman, A privacy enhanced authentication scheme for securing smart grid infrastructure, IEEE Trans. Ind. Inform. 18 (7) (2022) 5000–5006, http://dx.doi.org/10.1109/TII.2021.3119685.
[38] S.A. Chaudhry, K. Yahya, S. Garg, G. Kaddoum, M.M. Hassan, Y.B. Zikria, LAS-SG: An elliptic curve-based lightweight authentication scheme for smart grid environments, IEEE Trans. Ind. Inform. 19 (2) (2023) 1504–1511, http://dx.doi.org/10.1109/TII.2022.3158663.

[39] C. Cremers, Scyther, in: Semantics and Verification of Security Protocols, (Thesis), University Press Eindhoven, 2006.
[40] G. Lowe, A hierarchy of authentication specifications, in: Proceedings 10th Computer Security Foundations Workshop, IEEE, 1997, pp. 31–43.

Mohd Shariq received his B.Tech. degree in Computer Science and Engineering from Gautam Buddha Technical University, India, and his M.Tech. and Ph.D. degrees in Computer Science and Technology from the School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India, in 2017 and 2022, respectively. He is currently working as a Postdoctoral Fellow in the Department of Information Systems and Security at the College of Information Technology, United Arab Emirates University, Abu Dhabi, UAE. He has also worked as a Postdoctoral Fellow in the Department of Mathematics at the University of Padua, Italy, where he was part of the SPRITZ Research Group. His primary research interests include RFID security protocol design, security and privacy, lightweight RFID authentication, and information security. He has published several research papers in reputed journals. journals.

Mauro Conti received the Ph.D. degree from the Sapienza University of Rome, Italy, in 2009. He is a Full Professor with the University of Padua, Italy. He is also affiliated with TU Delft and the University of Washington, Seattle. After his Ph.D., he was a Postdoctoral Researcher with Vrije Universiteit Amsterdam, The Netherlands. In 2011, he joined as an Assistant Professor with the University of Padua, where he became an Associate Professor in 2015, and a Full Professor in 2018. He has been Visiting Researcher with GMU, UCLA, UCI, TU Darmstadt, UF, and FIU. His research is also funded by companies, including Cisco, Intel, and Huawei. His main research interest is in the area of security and privacy. In this area, he published more than 400 papers in topmost international peer-reviewed journals and conferences. He has been awarded the Marie Curie Fellowship by the European Commission in 2012, and a Fellowship by the German DAAD in 2013. He is an Area Editor-in-Chief for IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, and he has been an Associate Editor for several journals, including IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT. He was a Program Chair for TRUST 2015, ICISS 2016, WiSec 2017, and ACNS 2020, and a General Chair for SecureComm 2012, SACMAT 2013, CANS 2021, and ACNS 2022. He is a Senior Member of ACM. He is a member of the Blockchain Expert Panel of the Italian Government. He is Fellow of the Young Academy of Europe.

Karan Singh received the Engineering degree (Computer Science & Engineering) from Kamala Nehru Institute of Technology, Sultanpur, UP, India. He is the M.Tech. (Computer Science & Engineering) and Ph.D. (Computer Science & Engineering) from Motilal Nehru National Institute of Technology UP, India. He worked at Gautam Buddha University, UP, India. Currently, he is working with the School of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi. His primary research interests are in Computer network, Network security, Multicast communication, IoT, and Body Area network. He is the supervisor of many research scholars. He is a reviewer of Springer, Taylor & Francis, Elsevier Journals and IEEE Transactions. He is an Editorial Board Member of Journal of Communications and Network (CN), USA. He published 70+ research papers in refereed journals and good conferences. He organized the workshops, conference sessions, and training. Dr. Singh worked as General Chair of the international conference (Qshine 2013) at Gautam Buddha University, India. Recently he organized a conference ICCCS 2018 at Dronacharya College of Engineering, Gurgaon and a special session in 2nd ICGCET 2018 at Denmark.

Chhagan Lal received the Ph.D. degree in computer science and engineering from the Malaviya National Institute of Technology, Jaipur, India, in 2014. He is currently working as a Postdoctoral Research Fellow of the Delft University of Technology, The Netherlands. Previously, he was a Postdoctoral Fellow of the Department of Mathematics, University of Padua, Italy, where he was part of the SPRITZ Research Group. He was a Postdoctoral Research Fellow of Simula Research Laboratory, Norway. His current research areas include applications of blockchain technologies, security in software-defined networking, and Internet-of-Things networks. During his Ph.D., he was awarded the Canadian Commonwealth Scholarship under the Canadian Commonwealth Scholarship Program to work in the University of Saskatchewan, Saskatoon, SK, Canada.

Ashok Kumar Das received a Ph.D. degree in computer science and engineering, an M.Tech. degree in computer science and data processing, and an M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently a full professor with the Center for Security, Theory and Algorithmic Research, IIIT, Hyderabad, India. He is an adjunct professor at the Korea University, South Korea. He also worked as a visiting research professor with the Virginia Modeling, Analysis, and Simulation Center, Old Dominion University, Suffolk, VA 23435, USA. His current research interests include cryptography, system and network security, including security in smart grids, Internet of Things (IoT), Internet of Drones (IoD), Internet of Vehicles (IoV), Cyber-Physical Systems (CPS), and cloud computing, intrusion detection, Blockchain, AI/ML security, and post-quantum cryptography. He has authored over 425 papers in international journals and conferences in the above areas, including over 360 reputed journal papers. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He has been listed in the Web of Science (Clarivate™) Highly Cited Researcher 2022 and 2023 in recognition of his exceptional research performance. He was/is on the editorial board of IEEE Transactions on Information Forensics and Security, IEEE Systems Journal, Journal of Network and Computer Applications (Elsevier), Computer Communications (Elsevier), Journal of Cloud Computing (Springer), Cyber Security and Applications (Elsevier), IET Communications, KSII Transactions on Internet and Information Systems, and International Journal of Internet Technology and Secured Transactions (Inderscience). He also served as one of the Technical Program Committee Chairs of the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, June 2019, International Conference on Applied Soft Computing and Communication Networks (ACN'20), October 2020, Chennai, India, second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, October 2020, and International Conference on Applied Soft Computing and Communication Networks (ACN'23), December 2023, Bangalore, India. His Google Scholar h-index is 87, and his i10-index is 270, with over 22,600 citations. He is a senior member of the IEEE.

Shehzad Ashraf Chaudhry is working as an Associate Professor of Cybersecurity Engineering in College of Engineering, Abu Dhabi University, United Arab Emirates and also an Associate Professor of Software Engineering in Nisantasi University, Istanbul, Turkey. Dr. Shehzad received the master's and Ph.D. degrees (with Distinction) from International Islamic University Islamabad, Pakistan, in 2009 and 2016, respectively. He was a recipient of the Gold Medal for achieving 4.0/4.0 CGPA in his Masters. Prior to joining ADU he served as an Associate Professor with Istanbul Gelisim University, Istanbul, Turkey, University of Sialkot, Pakistan and as an Assistant Professor with International Islamic University, Islamabad, Pakistan. He has authored over 170 scientific publications appeared in different international journals and proceedings, including more than 140 publications in web of science journals. Some of his findings are published in top rated prestigious journals and conferences including IEEE Communications Standards Magazine, IEEE

Transactions on Industrial Informatics, IEEE Transactions on Vehicular Technology, IEEE Internet of Things Journal, IEEE Transactions on Intelligent Transportation Systems, IEEE Transactions on Industry Applications, IEEE Transactions on Reliability, ACM Transactions on Internet Technology, Sustainable Cities and Society (Elsevier), FGCS, IJEPES, Computer Networks, Computer Communications, and Digital Communications and Networks.

With an H-index of 48, an I-10 index 119, and accumulate impact factor of more than 490, his work has been cited over 6400 times. He has also supervised over 40 graduate students in their research. His current research interests include lightweight cryptography, elliptic curve cryptography, multimedia security, E-payment systems, MANETs, SIP authentication, Smart Grid security, IoT and Cloud infrastructure security, and next generation networks. He delivered several keynote speeches and holds several awards for best research papers. Considering his research, Pakistan Council for Science and Technology granted him the Prestigious Research Productivity Award, while affirming him among Top Productive Computer Scientists in Pakistan. For the consecutive four years (i.e., 2020 to 2023), he is being listed among the top 2% computer scientists across the world in Stanford University's report.

Mehedi Masud (Senior Member, IEEE) is a Professor in the Department of Computer Science at Taif University, Taif, Kingdom of Saudi Arabia. Dr. Mehedi Masud received his Ph.D. in Computer Science from the University of Ottawa, Canada. His research interests include machine learning, distributed algorithms, data security, formal methods, and health analytics. He has authored and co-authored around 100 publications, including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, books, and book chapters. He has served as a technical program committee member at different international conferences. He is a recipient of many awards, including the Research in Excellence Award from Taif University. He is on the Associate Editorial Board of IEEE Access, International Journal of Knowledge Society Research (IJKSR), and editorial board member of the Journal of Software. He also served as a guest editor of ComSIS Journal and Journal of Universal Computer Science (JUCS). Dr. Mehedi is a Senior Member of IEEE, a member of ACM.