# An Ultralightweight Mutual Authentication Protocol for Low Cost RFID Tags

R.K.Pateriya
Computer Science Department
Maulana Azad National Institute of Technology
Bhopal, India

Sangeeta Sharma
Computer Science Department
Maulana Azad National Institute of Technology
Bhopal, India

## ABSTRACT

RFID (Radio Frequency identification) is emerging as an important tool in the field of Automatic Identification Technologies. The universal deployment of RFID devices may expose new security and privacy risks. These risks are the main obstacle for successful deployment of RFID tags. Since, the traditional cryptographic approach is not suitable for the RFID tags due to its limited computation resources and small storage capacity. This paper proposes an effective and efficient ultralightweight mutual authentication protocol, to achieve stronger security and privacy by using only simple bitwise operations (e.g. XOR, modulus addition). The proposed protocol is inspired by the Gossamer protocol of ultralightweight protocol family and by using its existing Rotation and Mixbit function. The proposed protocol provides better solution for security and privacy risk as compared with Gossamer and other relevant protocols. The comparative security and performance analysis shows that proposed protocol provides better security and privacy from the other solutions as well as reduces the computation, communication and storage cost.

## Keywords

Authentication, Traceability, Applicability, Ultra-lightweight protocol, privacy, radio frequency identification (RFID), security, active attacks, denial of service, de-synchronization.

## 1. INTRODUCTION

Radio Frequency Identification (RFID) is a more specific category which comes under Automatic Identification Technologies. RFID is a system in which an object is uniquely identified by transmitting its identity (a unique serial number) through radio waves. RFID system works well in harsh or dirty environment, without the need for line of sight, whereas other Automatic Identification Technologies like bar codes fails to operate in such environment. RFID provides an easy way to collect information about a product, place, time or transaction quickly and without any human error.

### 1.1 Components of an RFID system

RFID system consist of various components [1] by which, it can identify objects (tag) and perform various operations on it. The integration of RFID components enables the implementation of an RFID solution. The RFID system consists of following four components as shown in Figure 1.

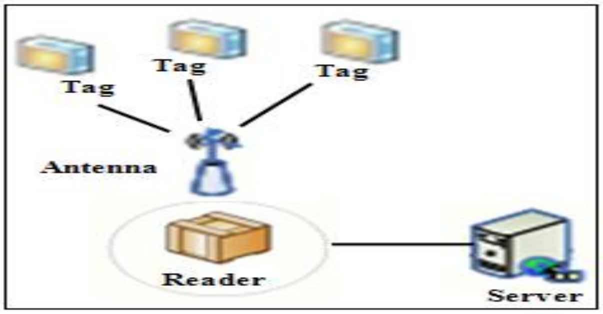


**Fig 1: Components of an RFID system**

- Tag: It is attached with an object, with its unique identification number.
- Antenna: It acts as a tag detector, which creates magnetic field.
- Reader: It works as a receiver of tag information.
- Backend Database Server: It is a user database server/application/ interface.

## Classification of Tags

In RFID system each object is identified by its unique ID, which is usually stored in tags. Tag is a device which contains microchips to store unique identification number of an object. The microchip is typical integrated circuit which is embedded on silicon chip. The stored ID can be permanent or changeable depending on the read and write characteristics of microchip.

Basically tags are classified into three categories; passive, active and semi active. Shortly, active tags are those which contains partial or complete battery, passive tags are those which doesn’t contain battery and semi active tags are combination of active and passive. The functions of tags to the RFID system are based on their characteristics like frequency range, read range, memory, data and security. RFID performance mainly depends on these tag characteristics.

### 1.2 Security issues in RFID System

In RFID system, communication channel between tag and reader is wireless which is vulnerable to various attacks. The main risks of the RFID system are as follows:

### 1.2.1 Mutual authentication

The basic need of RFID system security is that the communication takes place between valid tag and reader. For this the tag should authenticate reader in order to determine whether tag is sending its information to valid reader, and vice versa.

### 1.2.2 Tag anonymity

RFID tags along with unique identification number may also store user's privacy information, such as name, age, location, etc. This information should not be leaked and for this, tag anonymity should be preserved.

### 1.2.3 Untraceability

By analyzing the previous tag responses an adversary can not be able to identify tag and its location.

### 1.2.4 Forward security

Forward security ensures that even the current confidential information is obtained by an adversary, then also it would not able to get the previous confidential information.

### 1.2.5 Replay/Spoofing attack:

In this, attacker obtains the response of the tag by eavesdropping and sends it in the next authentication phase to achieve authentication.

### 1.2.6 Man in the Middle Attack/ Relay Attack:

This attack is a form of active eavesdropping. In this, the attacker works between the tag and reader as an interface and gives the illusion to them that they directly communicating to each other, when in fact the entire conversation is controlled by the attacker. The attacker easily intercepts all messages going between the tag and reader and also injects new ones, in wireless channel when attacker is in reception range of RFID system. In this way attacker affect the security and privacy of RFID system.

### 1.2.7 Desynchronization:

In a desynchronization attack the adversary try to disorder the protocol sequence and makes the information of tag and reader inconsistent. Due to this further authentication is not possible.

### 1.2.8 Denial of Service:

A denial-of-service attack (DoS attack) is an attempt to make a resource unavailable to its intended users. Since tag is resource constrained so, it is much vulnerable to DoS attack. Not only tag, reader also affected by this attack. Memory and Computational Exhaustive attack on tag which is counterpart of DoS attack in which tag unnecessary busy in computation in response to the query message send by the attacker.

## 1.3 Challenges and approaches to solve security issues in RFID

RFIDs can range from high cost to low cost. The low cost RFIDs are very resource constraint. Only 250-3K logic gates can be devoted to security related tasks out of total of 5K-10K logic gates. The standard for such resource constraint RFID is the EPCglobal Class-1 Generation-2.

Several approaches have been defined and classified as:-

- Full-fledged approach employing cryptographic functions and public key algorithms, to provide security and privacy.
- Simple approach can support random number generators and one-way hash functions.
- Lightweight approach supports a random number generation and simple functions like a Cyclic Redundancy Code (CRC) checksum, but not cryptographic hash function.
- "Ultralightweight" approach can only compute simple bitwise operations like XOR, AND, OR, etc.

RFID tag has very limited computational resources hence it can only employ lightweight approaches, which uses XOR, hash function etc., instead of symmetric or asymmetric encryption algorithm. But still, lightweight approach is not sufficient because it is costly as well as less secure. So, there is a need for ultralightweight approach but, it is a challenging task for many researchers to design ultrlightweight security protocols.

The rest of this paper is organized as follows. Section II, provides a review of the related works on RFID security and privacy. Then, Section III describes proposed ultralightweight mutual authentication protocol. The security and performance of proposed algorithm is analyzed in Section IV and compared with other schemes. Finally, Section V concludes the paper and point out future directions.

## 2. RELATED WORK

To address the security and privacy issues of RFID system, many counter measures have been proposed: Physical approaches [2] i.e. tag killing, tag sleeping, blocking, soft blocking, proxying etc. are the approaches basically deals with the privacy issues of the RFID system. Since RFID used in identifying products, people buying such products are also in danger of being traced. To resolve this, in tag killing approach, after sale the product tag is permanently deactivate or kill. But, using this all post purchased details is also lost. To overcome this problem tag sleeping is used, in which instead of permanently killing, tag is temporary deactivate. Another approach is blocking in which it uses specific tag called blocker tag, that interferes with readers to prevent unwanted scanning of tag. One of its variant is soft blocking. To deal with privacy, proxying approach includes another privacy enforcing device for

RFID**.** There are also many more physical approaches but none of them able to resolve security and privacy issues successfully.

Other method is cryptographic approach which provides various lightweight solution of on-tag access control and tag-reader authentication. Until now, the hash-lock, randomized hash lock, the hash chain [3] and the challenge-response based RFID protocols and so on, are all based on hash function. He Lei et al. [4] gives the analysis of a One-way Hash based Low-cost authentication protocol and finds its weakness which provides better solution to remove its weakness and also provide better security and performance as compare to other schemes.

Due to hash function uninvertible characteristics it is a good candidate for low-cost tag design. But, the cost of a hash function is still higher than the basic operations. Due to cost factor other lightweight solution is also proposed which is shared pseudonym based and CRC-based protocols. Younghao Gu et al. [5] proposed a lightweight mutual authentication protocol based on CRC and XOR and also gives the comparison with other scheme and shows better results. Another lightweight mutual authentication protocol, which is based on shrinking generator, that can be considered an alternative to one time Pad algorithm proposed by Shemaili et al. [6], provides low- cost solution to the RFID passive tags. But still these approaches are not cost effective, so the researchers look towards ultralightweight solution which is cost effective and also resolve security issues of RFID.

In this class Peris et al. proposed a family of Ultralightweight Mutual Authentication Protocols. Chronologically, Minimalist Mutual Authentication Protocol ($M^2AP$) was the first proposal [7] in the family. This protocol had some weaknesses and was attacked in next year [8]. The next protocol was Lightweight Mutual Authentication Protocol (LMAP) [9], which was also attacked [10]. Efficient Mutual Authentication Protocol (EMAP) [11] was an enhanced version of LMAP, which also had some vulnerabilities [12].

These protocols guarantee tag anonymity with the use of pseudonyms. To retrieve the information associated from a tag (tag identification phase), an index-pseudonym is used by an authorized reader. The shared secret keys are used by both readers and tags to build the messages exchanged in the mutual authentication phase. In these protocols only bitwise operations like XOR, bitwise AND, bitwise OR and addition mod $2^m$ are used. On the other side only reader needs to generate pseudorandom numbers. Tags only use them to build the message to the protocol.

These proposed schemes consist of three phases. First identification phase in which the tag is identified by means of the index-pseudonym. Second is Authentication in which the reader and the tag are mutually authenticated and also used to transmit the static tag identifier (ID) securely. Finally the Updating phase in which the index-pseudonym and shared secret keys are updated (for details refer original papers).

Hung-Yu Chien proposed Strong Authentication and Strong Integrity (SASI) [13] protocol which overcame the vulnerabilities of EMAP. This protocol incorporates the first non-triangular rotation function, which was its main strength. The rotation function provided good diffusion properties. This protocol was also attacked and its vulnerabilities were uncovered [14] and [15]. To determine secret values SASI used XOR with addition modulo and an OR functions with known public value of IDS.

Still all these protocols were not strong enough and then Peris et al. proposed a Gossamer ultra-lightweight protocol [16]. This protocol uses two non-triangular functions including RotBits and MixBits which provided good confusion and diffusion properties. It also uses addition and XOR operations to prevent a divide and conquer attack launched on earlier versions.

Bilal, Masood and Kausar [17] present a security analysis of Gossamer protocol. It also propose a new mutual authentication protocol which can remove the possible vulnerabilities discovered in Gossamer protocol like denial of service, memory and computation exhaustive, de-synchronization, replay attack .

The protocol present in this paper is a pure ultralightweight mutual authentication protocol, which only uses basic operations such as XOR, AND, Rotation, Mixbit function. It needs less storage and computation cost, so easily applicable to resource constrained low-cost RFID System. It provides better security than the similar constrained environment protocols. It provides protection against all possible attacks either active or passive. Such as DoS, memory and computation exhaustive, de-synchronization, replay, IDS (index pseudonym) collision, and Tag anonymity, spoofing attack, Man in the middle attack forward security etc.

The proposed protocol may count as the next protocol in ultralightweight family. This protocol only uses bitwise operation, not even reader needed random number generation. It provides better security as well as more efficiency in terms of storage, communication and computation cost.

## 3. PROPOSED PROTOCOL

An ultralightweight RFID mutual Authentication protocol proposed in this section, which uses only bitwise operation to resolve the security and privacy issues for low- cost RFID tag.

### 3.1 Preliminaries

In the protocol a general assumption for an RFID system model is that it consists of main three components: tags, readers, and a back-end server. Communication has to be initiated by readers, since tags are passive. The communication channel between the reader and the back-end database is secure in the sense that some advanced encryption technique is used. But the reader communicates with the tag through wireless channel which is insecure and can be eavesdrop by adversary. The notations used in the proposed protocol are as follows:

- T: RFID Tag
- R: RFID Reader
- DBS: Back-end Database Server
- ID: Unique and static identification information stored in each tag.
- IDS: index-pseudonym is a dynamic identifier employed as a search index to allocate, in the database, all the information linked with each tag.
- k1 and k2: two secret keys stored in tag memory and also stored in the back-end database.
- ⊕: indicates bitwise XOR operation.
- +: addition mod $2^m$
- Rot(x,y) : The function performs left rotation by circular shift on the value of x, (y mod N) positions to the left for a given value of N (in our case 96). [16]
- *A*→ *B* refers to assigning A to B.
- MixBits(X,Y): Random number generation is a costly operation, increases both memory requirements and message counts. To significantly increase security, protocol MixBit[16] is used which is highly non-linear and very lightweight function, as only bitwise right shift (>>) and additions are employed. Specifically,

  ```
  Z = MixBits(X,Y)
  ----------------------------
  Z = X;
  for(i=0; i<32; i++) {
  Z = (Z>>1) + Z + Z + Y ;}
  ----------------------------
  ```

All parameters in the protocol are of length 96-bit compatible with all the encoding schemes (i.e. GTIN, GRAI) defined by EPCGlobal.

## 3.2 The Proposed Protocol

The protocol comprises three stages: tag identification phase, mutual authentication phase, and updating phase. Figure 2 shows the exchanged messages.

### 3.2.1 *Tag Identification*

The reader first sends a "hello" message to the tag. Then the tag responds with message M, which consist of public message A and index- Pseudonym (IDS). The reader uses this IDS as a reference to search for the shared keys of the tag in its database and by using its secret keys, addition operation, ROT and MIXBIT function tag generate public message "A". This message is used by reader in the next step of mutual authentication. The secret ID is also sends with the message A. The database stores the private information in a pair of old values and updated new values after each successful protocol run. And it will use to avoid desynchronization attack. Reader compare the received IDS with new values and if it will not matched than compare with old values, if database has an entry against an IDS means tag is valid and move to next authentication phase. Otherwise protocol Stops.

### 3.2.2 *Mutual Authentication*

With IDS, the reader acquires the private information linked to the tag, identified from the database. Then by using private information reader compute the local version of A as A'. In response it sends message B, which is used for reader authentication and also updating confirmation.

*Tag Authentication:* On receiving message "A", this value is compared with a computed local version. If comparison is successful, the tag is authenticated; otherwise the protocol is abandoned.

*Reader Authentication:* On receiving message "B" from the reader, tag builds the local version of message B as B'. This is compared with the received value B. If both values are same, the reader is authenticated.

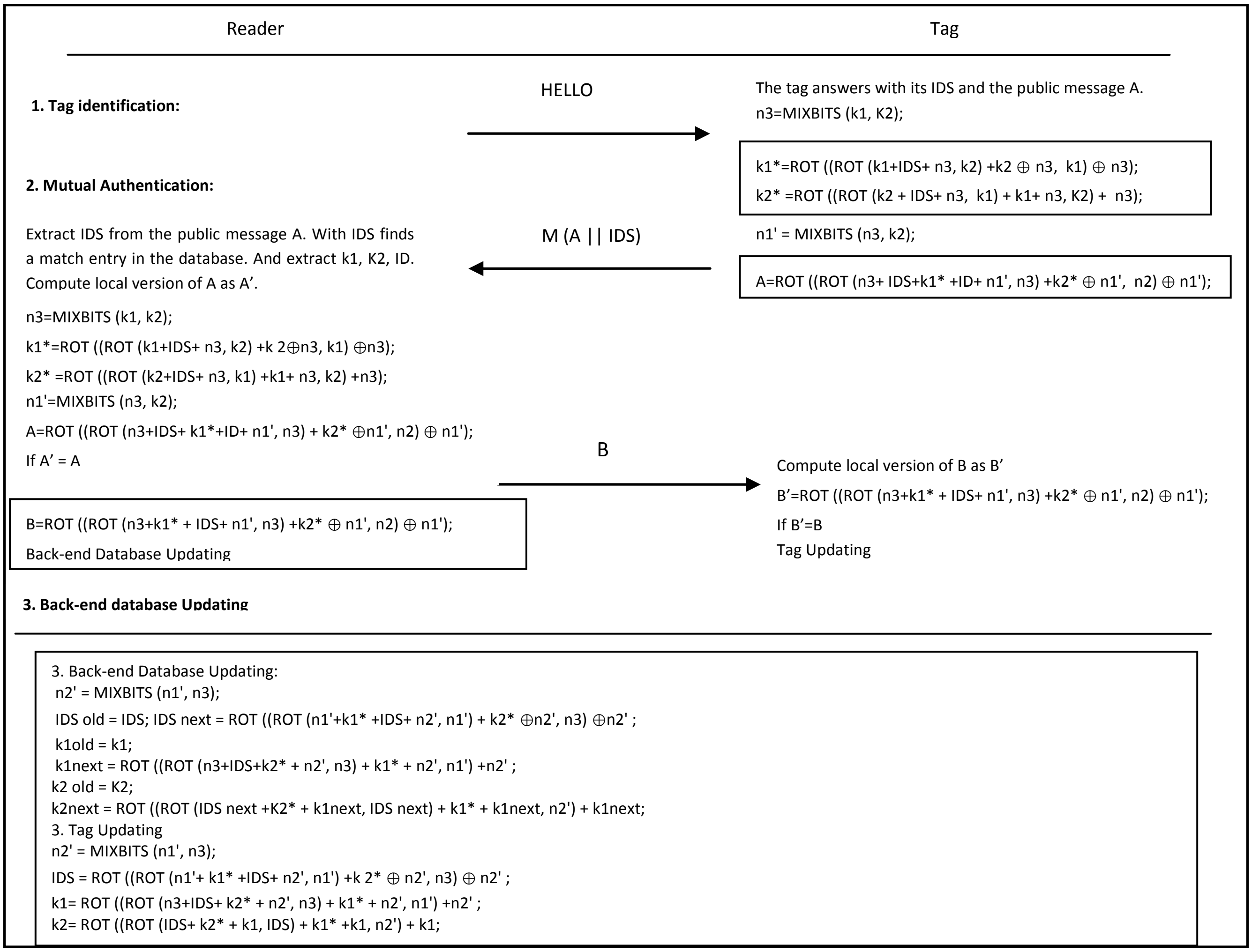
### 3.2.3 *Updating Phase*

After successful mutual authentication, both tag and the reader update their shared secrets as follows:

*Reader Updating:* After successfully completing the mutual authentication phase between the reader and the tag, the reader stores the older values and locally updates IDS and keys (k1, k2) as follows:

$$
\begin{aligned}
&n2' = \ MIXBITS(n1', n3);\\
&IDSold = IDS;\\
&IDSnext = ROT(ROT(n1' + k1^* + IDS + n2', n1') + k2^*\\
&\qquad\qquad \oplus n2', n3) \oplus n2';\\
&k1old = k1;\\
&k1next = ROT(ROT(n3 + IDS\\
&\qquad\qquad + k2^*\text{+n2',n3)+k1*+n2',n1')+n2'};\\
&k2old = k2;\\
&k2next = ROT(ROT(IDSnext\ + k2^* + k1next, IDSnext)\\
&\qquad\qquad + k1^* + k1next, n2') + k1next;
\end{aligned}
$$

Now, the reader checks whether a similar IDS already exists in its database. If updated IDS do not collide with existing entries, the Reader sends message B to allow the Tag to update values.



**1. Tag identification:**

The tag answers with its IDS and the public message A.
n3=MIXBITS (k1, K2);

k1*=ROT ((ROT (k1+IDS+ n3, k2) +k2 ⊕ n3, k1) ⊕ n3);
k2* =ROT ((ROT (k2 + IDS+ n3, k1) + k1+ n3, K2) + n3);

**2. Mutual Authentication:**

n1' = MIXBITS (n3, k2);

A=ROT ((ROT (n3+ IDS+k1* +ID+ n1', n3) +k2* ⊕ n1', n2) ⊕ n1');

Extract IDS from the public message A. With IDS finds a match entry in the database. And extract k1, K2, ID. Compute local version of A as A’.

n3=MIXBITS (k1, k2);
k1*=ROT ((ROT (k1+IDS+ n3, k2) +k 2⊕n3, k1) ⊕n3);
k2* =ROT ((ROT (k2+IDS+ n3, k1) +k1+ n3, k2) +n3);
n1'=MIXBITS (n3, k2);
A=ROT ((ROT (n3+IDS+ k1*+ID+ n1', n3) + k2* ⊕n1', n2) ⊕ n1');
If A’ = A

B=ROT ((ROT (n3+k1* + IDS+ n1', n3) +k2* ⊕ n1', n2) ⊕ n1');
Back-end Database Updating

Compute local version of B as B’
B’=ROT ((ROT (n3+k1* + IDS+ n1', n3) +k2* ⊕ n1', n2) ⊕ n1');
If B’=B
Tag Updating

**3. Back-end database Updating**

3. Back-end Database Updating:
n2' = MIXBITS (n1', n3);
IDS old = IDS; IDS next = ROT ((ROT (n1'+k1* +IDS+ n2', n1') + k2* ⊕n2', n3) ⊕n2' ;
k1old = k1;
k1next = ROT ((ROT (n3+IDS+k2* + n2', n3) + k1* + n2', n1') +n2' ;
k2 old = K2;
k2next = ROT ((ROT (IDS next +K2* + k1next, IDS next) + k1* + k1next, n2') + k1next;
3. Tag Updating
n2' = MIXBITS (n1', n3);
IDS = ROT ((ROT (n1'+ k1* +IDS+ n2', n1') +k 2* ⊕ n2', n3) ⊕ n2' ;
k1= ROT ((ROT (n3+IDS+ k2* + n2', n3) + k1* + n2', n1') +n2' ;
k2= ROT ((ROT (IDS+ k2* + k1, IDS) + k1* +k1, n2') + k1;

**Fig 2: Proposed Protocol**

In case, the IDS collide with existing values, the reader updates its values as:

$$n2' = MIXBITS(n3, n1');$$
$$IDSold = IDS;$$
$$IDSnext = ROT(ROT(n1' + k1* + IDS + n2', n1') + k2* \oplus n2', n3) \oplus n2';$$
$$k1old = k1;$$
$$k1next = ROT(ROT(n3 + IDS + k2*+n2',n3)+k1*+n2',n1')+n2';$$
$$k2old = k2;$$
$$k2next = ROT(ROT(IDSnext\ + k2* + k1next, IDSnext) + k1* + k1next, n2') + k1next;$$

Now, the reader sends B' to allow the tag to update its values.

*Tag Updating:* The tag after receiving an allow message B from the reader checks its legitimacy by computing a local value of B and comparing both. In case both value matches, it updates its values as:

$$n2' = MIXBITS(n1', n3);$$
$$IDS=ROT(ROT(n1'+k1*+IDS+n2',n1')+k2*\oplus n2',n3)\oplus n2';$$
$$k1 = ROT(ROT(n3 + IDS + k2* + n2', n3) + k1* + n2', n1') + n2';$$
$$k2 = ROT(ROT(IDS + k2* + k1, IDS) + k1* + k1, n2') + k1;$$

In case both values do not match, it computes a local value for B' for comparison. In case both values are equal it updates its values as:

$$n2' = MIXBITS(n3, n1');$$
$$IDS=ROT(ROT(n1'+k1*+IDS+n2',n1')+k2*\oplus n2',n3)\oplus n2';$$
$$k1 = ROT(ROT(n3 + IDS + k2* + n2', n3) + k1* + n2', n1') + n2';$$
$$k2 = ROT(ROT(IDS + k2* + k1, IDS) + k1* + k1, n2') + k1;$$

In case, either values are not compared successfully or the tag does not receive this update allow message, it does not update its values.

# 4. SECURITY ANALYSIS

We will now analyze the security of the proposed scheme against relevant attacks:

## 4.1 Mutual Authentication and Data Integrity

The protocol provides mutual authentication. Only a legitimate tag possessing keys (k1, k2) can build a valid message A. Similarly, only a genuine reader can obtain ID from A, and then compute message B.

Messages A and B, which involve the internal secret values (n3, n1', $k1^*$, $k2^*$) and keys (k1, k2), allow data integrity to be checked. Note that these values are included in the updating equations (potential next index-pseudonym and keys).

## 4.2 Data Confidentiality

All public messages are composed with four internal secret values (n3, n1', $k1^*$, $k2^*$) which is computed through private information (ID, k1, k2), which is shared only by legitimate readers and genuine tags. The static identifier and the secret keys cannot, therefore, be easily obtained by an eavesdropper.

## 4.3 Tag anonymity

After successful authentication each tag updates IDS and private keys (k1, k2), and this updation process involves random numbers (n3, n1', n2'). When the tag is interrogated again, a fresh IDS is backscattered. Additionally, all public submessages (A and B) are anonymized by the use of random numbers ( n3 , n1'). Tag anonymity is thus guaranteed, and location privacy of the tag owner is not compromised.

## 4.4 Forward Security

Forward security is the property that guarantees the security of past communications is not compromised even when at any time its secret information is disclosed. The attacker still cannot infer any information from previous sessions as five internal secret values (n3, n1', n2', k1*, k2*) are involved in the message creation (mutual authentication phase). Additionally, these internal values are employed in the updating phase. Consequently, past communications cannot be easily jeopardized.

## 4.5 Untraceability/Tracking attack

Tracking attack is the powerful attack which has both malicious active attackers as well as passive attackers. The goal of this attack is to track the specific tag by actively scan the response of tag. If the tag ever replies the same message twice, such as the same IDS, it can be traced. To avoid all these conditions, the proposed protocol updates the secret information after each successful protocol run. Therefore, an adversary cannot make a link between a tag and its response. The proposed protocol also avoids the problem of duplicate IDS by preventing the IDS collision in the database. So, the proposed protocol is secure from tracking.

## 4.6 Updating Confirmation

In the proposed protocol, updating confirmation in form of message B is send to the tag by the reader and after verification of message B tag updates itself and thus de-synchronization attack is avoided. Reader stores two values (previous, new) of shared secret information. By any means if the tag will not update its secret values in that case, as the reader is keeping older values of IDS and keys, the reader will recognize the tag by its older values and both cannot fall out of synchronization.

## 4.7 Man-in- the-middle attack

In this, an attacker can act as the middle man between the tag and reader to seize the exchanged messages. Where as, in the proposed protocol encrypted messages are used, so the middleman will not be able to understand the messages.

Only way to decode the message by the attacker is to obtain the secret information by some physical attack. However, the private information stored in tag chip is password protected, which reduces the chance of physical attack. Moreover, the tag's holder should also aware of such attacks and must not show tag to any unauthorized person.

## 4.8 Message Replay and De-Synchronization Attacks

An eavesdropper could store all the messages exchanged in a protocol run. In the next session it replays with these messages and try to deduce the private information and causes de-synchronization between the tag and the reader. The proposed protocol avoids both replay and desyncronization attack. To overcome from replay attack, private information is updated in each session and to avoid desynchronization attack, tag update itself after getting shared secret as confirmation from the reader. Shared secrets will ensure that this message is sent by the legitimate reader. This will prevent the tag updating without ensuring whether messages A was verified by the reader or not. If A is verified correctly, message B is sent, otherwise a message to abandon protocol is sent and the tag may not change its internal state, thus avoiding de-synchronization. As reader stores two values (previous, new) of private information, if the attacker stops the message B, in that case reader has been updated but the tag has not been updated yet. In this scenario, the tag is identified by the old index-pseudonym and the attacker may forward the eavesdropped values of A and B. Even if this is successful and some internal state is changed in the genuine tag, no secret information is disclosed, so all these attacks are unsuccessful.

## 4.9 Denial of Service Attacks

The proposed protocol also provides protection against a DoS attack, which is an active attack. For this problem the protocol uses the simple solution given in [17]. So, by incorporating a counter in the tag, DoS attack and its variant memory and computation exhaustive attacks can be avoided. Similarly, a kind of DoS attack on the reader exploiting its weakness to re-communicate in case of a backscattered IDS not recognized is avoided in the proposed scheme.

To overcome the weakness of a reader and DoS attack, a counter may be used with each hello message and incremented with its reply by the tag while sending IDS. An overflow condition can be employed depending on the reliability of the network connection. If the counter reaches to threshold level it depicts that tag was accessed many times but complete protocol was not accomplished so far. It will be an indication to a DoS attack. The tag can now stop responding to further request for some period of time and after that counter may reset. This counter will also reset only once when keys and IDS updating stage is reached which ensures that protocol is successfully accomplished. Similarly to avoid DoS attack same policy can be introduced in more powerful reader. It is already explain that the tag update does not take place until allowed by the reader. This ultimately avoids an IDS not recognizable, hence, preventing a DoS attack of such kind.

## 4.10 IDS Collision in Database

Scheme presented in [17] addresses the issues of IDS collision in the database and provide solution to avoid problem. By using its idea the proposed protocol also able to avoid IDS collision in the database. The solution to this problem is that, if such a situation arises, the reader may change the update equation and send an indication to the tag by sending B' instead of message B. This message is used to generate an IDS which does not collide with existing entries. The tag can now verify B and in case of failure, tries to verify B ' with its locally computed version. Now, the reader has ensured that this new updated value does not collide with any of the existing IDS values which although is very unlikely.

# 5. PERFORMANCE ANALYSIS

Proposed protocol is now analyzed in terms of computational cost, storage requirements and communication cost. Additionally, Table 1. compares the most relevant ultralightweight protocol proposals from a performance perspective.

## 5.1 Computational cost

The proposed protocol only requires simple bitwise XOR, addition 2m, left rotation, and the MixBits function on tags. These operations are very low-cost and can be efficiently implemented in hardware. Since tag has limited computation power, so the computational cost of the proposed protocol on tag is calculated as how many bitwise operations are used by tag for a complete run. The proposed protocol required less bitwise calculations than other relevant protocol. Since, the proposed protocol uses only two messages for authentication and as a result number of calls for bitwise operation is also minimized.

On the database side with the similar computation as that on the tags, during interrogation it needs to search the database to compare IDS and to extract the private information. By using appropriate searching algorithm, the server could find the match with complexity of O(1) and in batch mode O(n).

## 5.2 Storage requirement

Each tag stores its static identifier (ID) and record of the tuple (IDS, k 1, k 2) values. A 96-bit length is assumed for all elements in accordance with EPCGlobal. The ID is a static value, thus stored in ROM. The remaining values (96 × 3 = 288 bits) are stored in a rewritable memory because they need to be updated.

In the proposed protocol an additional value is derived from the two shared secret keys (i.e. n3 = MixBits(k1, k2 )) and it also updated in the internal steps of the protocol. For this, there is a need for additional memory which is incorporated in the algorithm logic unit (ALU) of the tag to temporary hold the values. So, with the relatively light penalty of this additional temporary memory, the security level seems to be remarkably increased.

## 5.3 Communication cost

The proposed protocol performs mutual authentication and integrity protection with only three messages. In the identification and authentication phase, a “hello” message of 40 bits and Messages A|| IDS and B of 192 bits are sent over the channel. So a total of 232 bits are sent over the channel. So it is clear that it provides lowest communication cost than all other relevant schemes.

# 6. CONCLUSIONS

Providing security and privacy to RFID system while, keeping low cost is the main issue towards the wide deployment of RFID. Since, RFID tags are low-cost and resource constrained, it is incapable of performing classical cryptographic operations.

Some researchers have provided many lightweight and ultralightweight protocols that could be implemented in low-cost tags. But none of them fulfills the security requirement of the low cost RFID system efficiently. This paper presents an efficient and secure ultralightweight RFID mutual authentication protocol using only basic bitwise operations.

The intensive security analysis shows that the new protocol can resist from most of the passive as well as active attack like spoofing attack. replay attack, man-in-the middle attack and other common attacks against RFID authentication protocol. It does not leak user’s confidential information and also provides forward security and prevents against desynchronization attack.

**Table 1. Comparison of Ultralightweight Protocols**

| **Security Features** | **U-MAP family [7,9,11]** | **SASI [13]** | **Gossamer[16]** | **Proposed Scheme** |
|---|---|---|---|---|
| Data Confidentiality | Yes | Yes | Yes | Yes |
| Privacy and Tag Anonymity | Yes | Yes | Yes | Yes |
| Mutual Authentication and Data Integrity | Yes | Yes | Yes | Yes |
| Forward Security | Yes | Yes | Yes | Yes |
| Resistance to De-synchronization and Replay Attacks | No | No | No | Yes |
| Updating Confirmation | No | No | No | Yes |
| Resistance to DoS Attacks | No | No | No | Yes |
| Resistance to IDS Collision | No | No | No | Yes |
| Total Messages for Mutual Authentication | 4-5L | 4L | 4L | 2L |
| Memory Size on Tag | 6L | 7L | 7L | 4L |
| Memory Size for each Tag on Database | 6L | 4L | 4L | 7L |
| Operation Types on Tag | AND, ⊕, OR, +, | AND, ⊕, OR, +, Rot | ⊕, +, Rot, MixBits | ⊕, +, Rot, MixBits |

Moreover, the efficiency evaluation shows that the new protocol is low-cost, in terms of storage, communication and computation, which fits the limited calculation capacity and storage demand of tag. The protocol is also practical so as to be applied in many security sensitive RFID applications. Hence, the proposed work met the challenge on designing secure and ultralightweight RFID mutual authentication protocol under low-cost RFID application scenario. The future work may include, increasing the overall complexity for breaking the protocol by keeping other parameters as low as possible and also try to resist from physical attacks.

## 7. REFERENCES

[1] Kamran Ahsan, Hanifa shah, Paul Kingston "RFID Applications: An Introductory and Exploratory study," IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 1, No. 3, January 2010.

[2] Juels, "RFID security and privacy: A research survey," IEEE Journal on Selected Areas in Communication, vol. 24, no.2, pp.381– 394, February 2006.

[3] Dong-Her Shih Chin-Yi Liṇ Binshan Liṇ" Privacy and Security Aspects of RFID Tags," International Journal of Mobile Communications - Vol. 3, No 3 pp. 214 – 230, 2005.

[4] He Lei,Lu Xin-mei,Jin Song-he,Cai Zeng-yu "A one-way Hash based low-cost authentication protocol with forward security in RFID system ," in Proc. 2nd International Asia Conference on , Informatics in Control, Automation and Robotics (CAR), 2010, pp. 269 – 272.

[5] Yonghao Gu, Weiming Wu "Mutual authentication protocol based on tag ID number updating for low-cost RFID," in proc. IEEE International Conference on Network Infrastructure and Digital Content, 2009, pp. 548 – 551.

[6] Shemaili, M.A.B., Chan Yeob Yeun, Zemerly, M.J. "RFID lightweight mutual authentication using shrinking generator," in Proc. International Conference for Internet Technology and Secured Transactions, 2009. ICITST 2009, pp. 1 – 6.

[7] P. Peris-Lopez, J. C. H. Castro, J. M. Est´evez-Tapiador, and A. Ribagorda, "M2AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags," in Proc. UIC, 2006, pp. 912–923.

[8] M. B´ar´asz, B. Boros, P. Ligeti, K. L´oja, and D. Nagy, "Passive Attack Against the M2AP Mutual Authentication Protocol for RFID Tags," in Proc. First International EURASIP Workshop on RFID Technology, Vienna, Austria, September 2007.

[9] P. Peris-lopez, J. C. Hern, J. M. E. Tapiador, and A. Ribagorda, "LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags," in Proc. 2nd Workshop on RFID Security. Ecrypt, 2006, ,p. 06.

[10] M. Barasz, B. Boros, P. Ligeti, K. Loja, and D. Nagy, "Breaking LMAP," in Proc. Conference on RFID Security, Malaga, Spain, July 2007.

[11] P. Peris-lopez, J. C. Hern, J. M. Estevez-tapiador, and A. Ribagorda, "EMAP: An efficient mutual authentication protocol for low-cost RFID tags," in Proc. OTM Federated Conferences and Workshop: IS Workshop. Springer-Verlag, 2006, pp. 352–361.

[12] T. Li and R. Deng, "Vulnerability Analysis of EMAP-An Efficient RFID Mutual Authentication Protocol," in Proc. International Conference on Availability, Reliability and Security, 2007, pp. 238–245.

[13] H.-Y. Chien, "SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity," IEEE Transactions on Dependable and Secure Computing, vol. 4, no. 4, pp.337–340, 2007.

[14] T. Cao, E. Bertino, and H. Lei, "Security Analysis of the SASI Protocol," IEEE Transactions on Dependable and Secure Computing, vol. 6, no. 1, pp. 73–77, 2009.

[15] H.-M. Sun, W.-C. Ting, and K.-H. Wang, "On the Security of Chien's Ultralightweight RFID Authentication Protocol," Cryptology ePrint Archive, Report 2008/083, 2008, http://eprint.iacr.org/.

[16] Peris-Lopez, Pedro, Hernandez-Castro, J. Cesar, Estevez-Tapiador, J. M., Ribagorda, and Arturo, "Advances in Ultralightweight Cryptography for Low-cost RFID Tags: Gossamer Protocol," in Proc. Workshop on Information Security Applications, ser. Lecture Notes in Computer Science. Jeju Island, Korea: Springer-Verlag, Sept. 2008.

[17] Bilal, Masood and Kausar "Security Analysis of Ultra-lightweight Cryptographic Protocol for Low-cost RFID Tags: Gossamer Protoco,l" 2 International Conference on Network-Based Information Systems, 2009.