22nd International Conference on Advanced Information Networking and Applications - Workshops

A Survey of RFID Authentication Protocols

Yawer Yousuf, Vidyasagar Potdar
Digital Ecosystems and Business Intelligence Institute, Curtin Business School, Curtin University of Technology, Perth, Western Australia
v.potdar@curtin.edu.au, yawer_yousuf@yahoo.com

Abstract

RFID are small wireless devices which can be used for identification of objects and humans as well. Their acceptance has grown in past years and is expected to grow further. Due to reduction in cost of production RFID devices are being deployed in large numbers in supply chains (by Wal-Mart, etc.) In this paper we provide a comprehensive survey of various RFID authentication protocols proposed in the literature and classify them in different categories. We then study RFID authentication protocols having minimalist technique namely EMAP, LMAP and M2MAP.

1. Introduction

RFID (Radio Frequency IDentification) is a technology used for the identification of objects. RFID has gained popularity in past few years. RFID technology started to replace the more tradition system of barcodes mainly due to the efforts of Wal-Mart, Procter and Gamble, etc.

A RFID system is basically composed of a RFID Transponder (tag) and a RFID Interrogator (Reader). The RFID tag is microchip connected to an antenna. This tag can be attached to an object, which needs to be uniquely identified, e.g. it can be used in a storehouse to track the entry and exit of goods. This tag contains information similar to the barcode, which stores the unique properties of the object to which it is attached. A RFID reader can access this information. The RFID reader communicates with the RFID tag using radio waves. The main advantage of RFID tags over barcode system is:

1. RFID system uniquely identifies the object "e.g. 114119201 is a bottle of jam of X company."
2. RFIDs do not require line of sight. The objects (tags) should be in a range much larger than barcodes would allow, and there is no need to individually scan each product.

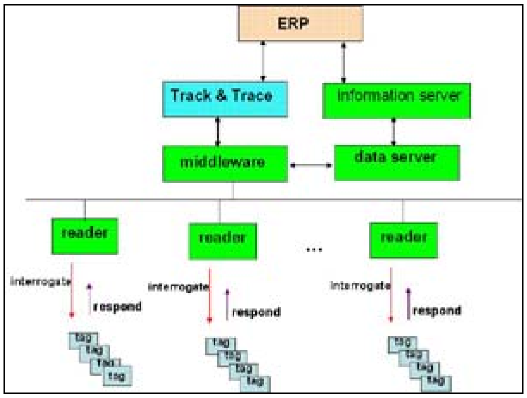

Figure 1:RFID Architecture [8]

RFID tags can be a *passive* tag which does not have any power source; they derive their power from the radio frequency generated by the reader. Tags that derive their own power are *semi-passive* tags in which batteries supply power when tags are interrogated by a reader and *passive* tags whose batteries provide power for transmission.

Within the RFID technology there are several security issues, which need to be tackled in order to make this technology more robust and reliable. The key security properties like confidentiality[1], integrity[2], availability, authentication and anonymity[3] need far more attention.

These security issues can be explained by the following scenario. Let us consider a storehouse, a malicious reader can eavesdrop the communication between tag and reader, thus *confidentiality* and *anonymity* is lost. A malicious reader can tamper the data stored in the tag, thereby compromising the data

[1] confidentiality in communication between the tag and the reader.
2 reliability of the information on the RFID tag.
3 Anonymity to undesired and anonymous scanning of items or people.

978-0-7695-3096-3/08 $25.00 © 2008 IEEE
DOI 10.1109/WAINA.2008.214

integrity. In some cases a message jamming attack or a Denial of Service attack can hamper the communication between a reader and a tag which can bring system to a halt by which current status at the storehouse cannot be made *available* at a moment. A malicious *authentication* can make a fake tag to impersonate the real one which can result in serious security issues.

In this paper we heavily focus on authentication issues and will provide a generic classification of various authentication protocols.. Authentication basically provides a certain level of trust amongst the reader and the tag such that the identity of the tag is verified and *vice versa.*

Each year quite a large number of RFID authentication protocols are published in scientific literature [2]. Some of these protocols are well-suited for only one particular solution, others are found to be fallacious and later corrected; and finally some proposals are trivial and are subsequently discarded. This induces us to give a proper classification of all the RFID authentication protocols. But attributes of a protocol such as its structure, or some complex cryptographic function may make classification difficult. Conceptually speaking classification means distinguishing on the basis of general prototypes which can cover various fundamental protocols. The author's in [1] stated that classification of authentication protocols is based on three points

1. Underlying algorithm used in the protocols.
2. Procedure of message exchange.
3. Secure combination of above two.

The concentration on message exchange has helped in abstracting away from cryptographic mechanism. There are few definitions which must be deduced from [1], these are discussed in detail in section 2 under preliminary concepts.

In section 3 we will explain the basic process of classification and the prominent prototypes of protocols. In section 4 we will discuss recent authentication protocols on RFID and analyze various security & privacy protection and integrity related issues. In the end we will conclude the paper in section 5.

2. Preliminary Concepts

Definition 1.*Forced Challenge (F):* If the fresh data is a random nonce generated by the verifier and then delivered as a plaintext or a ciphertext to the prover, then we say that the protocol uses a forced challenge to authenticate the prover.

Self Challenge(S): If the fresh data is generated by the prover himself the protocol is said to use self challenge.
No Challenge (Ø): When there is no challenge value exchanged in the protocol, we say that the protocol has no challenge.

Definition 2. *Origin Authentication (OA):* If a protocol contains a message which is generated by application of private key on cryptographic particles. i.e. the message is of the form $APriKey\{\bullet\}$ then we say the protocol provides origin authentication of the entity.

Destination Authentication (DA): If a protocol contains a message which is generated by application of public key on cryptographic particles. i.e. the message is of the form $APubKey\{\bullet\}$ then it provides destination authentication of the entity A.

Implicit Authentication (**IA**): If a protocol contains no message of the form $APriKey\{\bullet\}$ or $APubKey\{\bullet\}$, but still requires entity A to compute a value of the form $APriKey\{\ \bullet\}$, then we say that the protocol provides implicit authentication of A.

3. Protocol Classification

As discussed earlier as well as in [1], classification of authentication protocols implies distinguishing them on the basis of fundamental prototypes. However, the inclusion of extraneous information may make classification difficult. Therefore, the basic requirement is to identify the essential elements in the authentication protocols and the way they are combined and used. The authors have recognized the basic elements as the type of authentication and the types of challenge values. So the basic steps of classification are:

Step 1: Identify the type of authentication used in a given protocol. Is it Implicit Authentication (**IA**), Origin Authentication (**OA**) or Destination Authentication (**DA**)?

Step 2: Identify the type of challenge values used between two identities (i.e. sender and receiver) in a given protocol. is it forced challenge (**F**), self challenge (**S**) or no challenge (∅)?

Step 3: In case of DA with forced challenge, if there is responses by prover then the protocols are further classified into $DA_{F,\ No\ Ack}$ (No Acknowledgment) and $DA_{F,\ Ack}$.(Yes, Acknowledgment).

There are eight different prototypes for the classification and are summarized below as well as in Table 1:

3.1. Implicit Authentication

Implicit Authentication with no challenge ($\mathbf{IA}_{\emptyset}$): If the message does not contain any message of the form *APriKey*{•} or *ApubKey*{•}, but still requires entity *A* to compute a value of the form *ApriKey*{ •}. And no challenge value is exchanged between the identities. Then it is called Implicit Authentication with no challenge.

Implicit Authentication with forced challenge ($\mathbf{IA}_{\emptyset}$): If the message does not contain any of the form *APriKey*{•} or *APubKey*{•}, and requires entity *A* to compute a value of the form *APriKey*{ •}. In addition to that, the verifier computes random nonce generated by the verifier(through public or private key) and then sends it as a plaintext or cipher text. Then it is called Implicit Authentication with forced challenge.

3.2. Origin Authentication

Origin Authentication with no challenge ($\mathbf{OA}_{\emptyset}$): If the message contains the message of the form *APriKey*{•}, that is message is generated by applying private key and no challenge value is exchanged between the identities. Then it is called Origin Authentication with no challenge.

Origin Authentication with self challenge ($\mathbf{OA}_{S}$): If the message contains the message of the form *APriKey*{•}, and the data is generated at the prover end, then it is called Origin Authentication with self challenge[1].

Origin Authentication with forced challenge ($\boldsymbol{OA_F}$): If the message contains the message of the form *APriKey*{•}, and the data is generated by the verifier then it is called Origin Authentication with forced challenge.

3.3. Destination Authentication

Destination Authentication with no challenge ($\mathbf{DA}_{\square}$): If the message contains message of the form *APubKey*{•},and no challenge values is exchanged between the identities then is it called Destination Authentication with no challenge.

Destination Authentication with forced challenge ($\mathbf{DA}_{F}$): If the message contains message of the form *APubKey*{•}, and the verifier produces the random nonce then the authentication is called Destination Authentication with forced challenge. It can be further divided into two types.

1. *With Acknowledgment(*$\mathbf{DA}_{F,\ Ack}$*):* If the prover responds to the forced challenge by the verifier then the authentication is called Destination Authentication with forced challenge and acknowledgment.

2. *No Acknowledgment(*$\mathbf{DA}_{F,\ No\ Ack}$): If the prover does not respond to the forced challenge by the verifier then the authentication is called Destination Authentication with forced challenge and no acknowledgment.

Table 1 – Protocol Classification

Authentication Type		**Example**
Implicit Authentication (IA)	$IA_{Ø}$	A : *ApriKey*{ *B* }
	IA_F	$A \leftarrow B : r_B$ *A:*Ap*riKey* { B, r_B }
Origin Authentication (OA)	$OA_{\emptyset}$	$A \rightarrow B$: *APriKey*{ *B* }
	OA_S	$A \rightarrow B : TS_A$, *APriKey*{ B, TS_A }
	OA_F	$A \leftarrow B : r_B$ $A \rightarrow B$: *APriKey* { B, r_B }
Destination Authentication (DA)	$DA_{\emptyset}$	$A \leftarrow B$: *APubKey*{ *B* }
	$DA_{F,\ NoAck}$	$A \leftarrow B$: *APubKey*{ B, r_B }
	$DA_{F,\ Ack}$	$A \leftarrow B$: *APubKey*{ B, r_B } $A \rightarrow B : r_B$

3.4. Mutual Authentication

There should not be more than $8^2 = 64$ prototypes for mutual authentication by counting exhaustively. But the protocols in which, the responder entity B, act as an initiator can be regarded as *illegal.*

This condition rules out many prototypes which are mirror images of each other. The authors have identified 17 prototypes which come under illegal prototypes, so in all there are 47 (64-17) prototypes, which can be used for classification. The prominent protocols are summarized below in the Table 2.

Table 2

Prototype	Example
$IA_{F\text{-}\varnothing}$	1. A→B: r_A B: $BPriKey\{\ r_A\ \}$
$DA_{\varnothing}\text{–}\varnothing$	1. A →B: $BPubKey\{\ A\ \}$
$IA_{\varnothing}\text{–}IA_{\varnothing}$	A: $APriKey\{\ B\ \}$ B: $BPriKey\{\ A\ \}$
$IA_F\text{-}IA_F$	1. A →B: r_A 2. A ←B: r_B A: $APriKey\{\ B, r_B\ \}$ B: $BPriKey\{\ A, r_A\ \}$
$IA_F\text{-}OA_S$	1. A →B: r_A , TS_A , $APriKey\{\ B, TS_A\ \}$ B: $BPriKey\{\ r_A\ \}$
$OA_F\text{-}OA_F$	1. A →B: r_A 2. A ←B: $BPriKey\{\ A, r_A\ \}, r_B$ 3. A →B: $APrikey\{\ B, r_B\ \}$
$OA_F\text{-}DA_{F,NoAck}$	1. A →B: r_A 2. A ← B: $APubKey\{B, r_B , BPriKey\{\ A, r_A\ \}\ \}$ or, 1. A →B: r_A 2. A ← B: $BPriKey\{\ A, r_A , APubKey\{B, r_B\ \}\ \}$
$DA_{F,NoAck}\text{-}OA_S$	1. A →B: $BPubKey\{A, r_A , TS_A , APriKey\{\ B, TS_A\ \}\ \}$ or, 1. A →B: TSA , $APriKey\{\ B, TS_A , BPubKey\{A, r_A\ \}\ \}$
$DA_{F,Ack}\text{-}OA_F$	1. A →B: $BPubKey\{\ A, r_A\ \}$ 2. A ←B: r_A , r_B 3. A → B: $APriKey\{\ B, r_B\ \}$
$DA_{F,NoAck}\text{-}DA_{F,NoAck}$	1. A → B: $BPubKey\{\ A, r_A\ \}$ 2. A ←B: $APubKey\{\ B, r_B\ \}$
$DA_{F,Ack}\text{-}DA_{F,Ack}$	1. A →B: $BPubKey\{\ A, r_A\ \}$ 2. A ← B: $APubKey\{\ B, r_B\ \}, r_A$ 3. A → B: r_B

4. Discussion

4.1. Implicit Authentication with forced challenge- Implicit Authentication with forced challenge (IAF-IAF)

Minimalist cryptography approach: The real light-weight protocols were proposed by Pedro Peris-Lopez *et al.* namely, Lightweight Mutual Authentication Protocol (LMAP) [3] and Minimalist Mutual-Authentication Protocol (M2AP)[4] and Efficient Mutual Authentication Protocol (EMAP) [5]. In all three of the protocols simple binary operations like XOR, OR, AND, mod 2^m are used. Costly operation such as multiplication was not included. All the protocols are based on *index-pseudonyms* (96-bits) which is a row of a table to store all information related to the tag. It also uses a 480 EEPROM and a 96-bit key divided into 4 parts updates after each message cycle. Mutual Authentication is as follows:

Tag Identification: The reader sends a hello message to which tag responds by giving its IDS.

Reader Authentication: The reader generates random numbers n1 and n2 which are used to generate sub-messages A, B and C by using IDS and sub-keys K1, K2 and K3 respectively. The message A || B || C is transmitted to the tag where tag generates n1 and n2 which it uses to generate D. By the sub-messages A and B, the tag will authenticate reader.

Tag Authentication: Tag sends the sub-message D in case of LMAP and D and E in case of M2AP and EMAP containing the Static Identifier which in turn authenticates the tag. The whole authentication process is summarized in the table.

Reader Authentication	Tag Authentication
LMAP **Tag Identification Reader □ Tag:** ***hello*** **Tag □Reader: IDS**	

Reader □Tag: A||B||C
A = $IDS^{(n)}_{tag(i)}$ **XOR** $K1^{(n)}_{tag(i)}$ **XOR** n1
B = ($IDS^{(n)}_{tag(i)}$ **OR** $K2^{(n)}_{tag(i)}$) + n1
C = $IDS^{(n)}_{tag(i)}$ + $K3^{(n)}_{tag(i)}$ + n2
	Tag □Reader: D
	D = ($IDS^{(n)}_{tag(i)}$ + $ID_{tag(i)}$) **XOR** n1 **XOR** n2

M2MAP **Tag Identification – Similar to LMAP**	

A and C are same as LMAP
B = ($IDS^{(n)}_{tag(i)}$ ^ $K2^{(n)}_{tag(i)}$) **OR** n1
	Tag □Reader : D||E
	D = ($IDS^{(n)}_{tag(i)}$ **OR** $ID_{tag(i)}$) ^ n2
	E = ($IDS^{(n)}_{tag(i)}$ + $ID_{tag(i)}$) **XOR** n1

EMAP **Tag Identification – Similar to LMAP**	
A is same as LMAP B = ($IDS^{(n)}_{tag(i)}$ **OR** $K2^{(n)}_{tag(i)}$) **XOR** n1 $C = IDS^{(n)}_{tag(i)}$ ***XOR*** $K3^{(n)}_{tag(i)}$ ***XOR*** $n2$	Tag □ Reader : D||E $D = (IDS^{(n)}_{tag(i)}\ \ \hat{}K4^{(n)}_{tag(i)})$ ***XOR*** $n2$ $E = (IDS^{(n)}_{tag(i)}$ ^ $n1$ ***OR*** $n2)$***XOR***$ID_{tag(i)}\mathrm{M}^4_{I=1}KI^{(n)}_{tag(i)}$

4.2. Vulnerability of EMAP, LMAP and M2AP:

However, vulnerability of these protocols was identified by Tieyan Li *et al.* [5, 6, 7]. They showed the protocols were susceptible to attacks such as De-synchronization Attack such that they can not authenticate each other in any following protocol run and Full-Disclosure attack which can cause disclosure of all the information present in the tag including tag's ID. The countermeasures were proposed by build bit level error correcting mechanisms at the database and by sending a message Ď from tag irrespective of the authentication of reader. Both the cases will provide additional computation costs.

5. Conclusion

In this paper we studied several different RFID authentication protocols and focused on the three main researches i.e. EMAP, LMAP and M2MAP. We assert that other protocols can also be classified according to [1] to provide a more standardized study of RFID Authentication Protocols.

6. References

[1] DongGook Park, Colin Boyd, and Ed Dawson, "Classification of Authentication Protocols: A Practical Approach", *Proceedings of Information Security Workshop (ISW 2000), Springer-Verlag, LNCS Vol.1975*, pp.194-208

[2] Ari Juels, "RFID Security and Privacy: A research Survey", September 2005, *Manuscript, RSA Laboratories*, 2005.

[3] Peris-Lopez, Pedro and Hernandez-Castro, Julio Cesar and Estevez-Tapiador, Juan and Ribagorda, Arturo, "LMAP: A Real Lightweight Mutual Authentication Protocol for Low-cost RFID tags*" Printed handout of Workshop on RFID Security -- RFIDSec 06*, July 2006.

[4] Peris-Lopez, Pedro and Hernandez-Castro, Julio Cesar and Estevez-Tapiador, Juan and Ribagorda, Arturo,"M2AP: A Minimalist Mutual-Authentication Protocol for Low-cost RFID Tags", *Lecture Notes in Computer Science, 912--923, Springer-Verlag,* Sep-2006.

[5] Peris-Lopez, Pedro and Hernandez-Castro, Julio Cesar and Estevez-Tapiador, Juan M. and Ribagorda, Arturo, "EMAP: An Efficient Mutual Authentication Protocol for Low-cost RFID Tags", *OTM Federated Conferences and Workshop: IS Workshop -- IS'06, 2006, 4277 Lecture Notes in Computer Science*, P-352--361, November Springer-Verlag.

[6] Li, Tieyan and Wang, Guilin "Security Analysis of Two Ultra-Lightweight {RFID} Authentication" Protocols *IFIP SEC 2007*.

[7] Li, Tieyan and Deng, Robert H., "Vulnerability Analysis of {EMAP} - An Efficient RFID Mutual Authentication Protocols" *Second International Conference on Availability, Reliability and Security -- AReS 2007* April 2007 Vienna, Austria

[8] RFID Architecture Available Online - http://www.simtech.a-star.edu.sg/events/images/rg_RFID_BigSafe2.jpg Accessed on Friday, December 28, 2007