Light-weight RFID Tag-Reader Mutual Authentication Scheme

Divyan M. Konidala, Kwangjo Kim*

*Information and Communications Univ. (ICU),
International Research Center for Information Security (IRIS),
R504, 103-6, MunjiDong, YuseongGu, Daejeon 305732, Republic of Korea

Abstract

Cloned fake RFID tags and malicious RFID readers pose a major threat to the RFID-based supply chain management system. The data (*e.g.,* Electronic Product Code (EPC) number) on a genuine tag can be easily scanned and copied by a malicious RFID reader and the copied data can be embedded onto a fake tag. These cloned fake tags can be attached to counterfeit products, which can be introduced into a genuine supply chain, or illegally sold at black and grey markets. Malicious readers may also try to corrupt and snoop on genuine tags. These threats can only be nullified by incorporating a RFID tag-reader mutual authentication scheme. In this paper we propose a simple, secure, and light-weight tag-reader mutual authentication scheme that adheres to both EPCglobal Architecture Framework specification and EPCglobal Class 1 Gen 2 UHF RFID Protocol ratified standard. This scheme utilizes the tag's access password and also allows the manufacturer of the product to play a vital role in the tag-reader mutual authentication process. As a result we can achieve the following three goals: detect cloned fake tags, ward off maliciously snooping readers, and in the process, the manufacturer can also keep track on the whereabouts of its products, which have genuine tags attached to them.

I. Introduction

Radio Frequency Identification (RFID) has the capability to detect and identify objects using Radio-Frequency (RF) signal. It is anticipated to replace existing identification technologies like the bar code. With RFID technology, passive RFID tags are attached to objects/products and these tags contain tiny, but durable computer chips with very small antennas. Passive tags are powered-up from the interrogation Radio-Frequency (RF)signal of a reader. The tiny computer chips contain an Electronic Product Code (EPC) that uniquely identifies the object to which it is attached to, and the antennas automatically transmit this EPC number without requiring line-of-sight (*i.e.,* visual) scanning, to RFID readers within a certain RF range. Therefore RIFD technology allows quick scanning of products in large bulks.

RFID offers strategic advantages for businesses because it can track inventory in the supply chain more efficiently, and provide real-time products' track and trace capability. EPCglobal IncTM [1] is leading the development of industry-driven standards for the Electronic Product CodeTM (EPC) to support the use of Radio Frequency Identification (RFID) in supply chain management. VeriSign's white paper entitled "The EPCglobal Network: Enhancing the Supply Chain" [2] gives a detailed description about RFID technology and its advantages for supply chain management. Some of these advantages are as follows: RFID automates supply chain management, enabling

enterprises to realize significant savings to the top and bottom line. RFID technology greatly helps enterprises to maintain the accuracy of shipments sent and received by parties throughout distribution. It prevents product theft by capturing product arrival and departure at each point, enabling comprehensive distribution visibility that creates a record of the chain of custody for each product. The capability to pinpoint the custodian of the product when it was lost allows the manufacturer or retailer to take preventative measures for the future. As a result, it helps in precise product recall and prevents product counterfeiting.

We composed this paper based on the following specification and ratified standards from EPCglobal IncTM.

- EPCglobal Architecture Framework Version 1.0. [3]: This specification broadly defines the principles, standards, and components necessary to successfully develop and implement the EPCglobal Network, upon which trading partners or EPCglobal Subscribers will be able to rely to more efficiently manage their supply chain and operate their businesses.

- EPCTM Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz 960MHz Version 1.0.9. [4]:This EPCglobal Board Ratified standard defines the physical and logical requirements for a passive-backscatter, Interrogator-talks-first (ITF), radio-frequency identification (RFID) system operating in the 860 MHz - 960 MHz frequency range. The system comprises RFID Readers, and RFID Tags.

- EPCglobal Certificate Profile [5]:The authentication of entities (subscribers, services, physical devices) operating within the EPCglobal network serves as the foundation of any security function incorporated into the network. It is expected, however, that the X.509 authentication framework will be widely employed within the EPCglobal network. To ensure broad interoperability and rapid deployment while ensuring secure usage, this document defines a profile of X.509 certificate issuance and usage by entities in the EPCglobal network.

We carried out a through security assessment of the EPCglobal Architecture Framework in our paper titled "Security Assessment of EPCglobal Architecture Framework" [7]. The framework is composed of entities like RFID Tag, RFID Reader, RFID Middleware, Electronic Product Code Information Service (EPCIS) Repository, EPCIS Accessing Application, Object Naming Service, and Subscriber Authentication. We analyzed the various security threats that affect each of these entities and their communication interfaces. Some of these threats include cloned fake RFID tags, unauthorized access and/or modification of RFID tag information and its electronic pedigree (EPCIS data), and eavesdropping, spoofing and Denial of Service attack on EPCglobal Subscriber's network.

From our above-mentioned assessment we found that, the electronic pedigree of the items within the supply chain, and EPCglobal Subscriber's network can be protected by undertaking the following measures: "Subscriber Authentication" a core service of the framework can issue X.509 certificates and public-private security keys to the EPCglobal Subscribers and this helps in mutual authentication, authorization and establishing secure communication channels among the communicating EPCglobal Subscribers. Similarly all the resource rich entities like RFID reader, RFID Middleware, EPCIS Repository, EPCIS Accessing Application, and ONS can authenticate, authorize and establish secure communication channels by using X.509 Authentication Framework and technologies like SSL-TLS and EAP-TLS. We also need to protect application servers (RFID Middleware, EPCIS Accessing Application) and database servers (EPCIS Repository) by installing system authentication and role-based access control, firewall, intrusion detection system, anti-virus software, and input data and SQL query validation. But the threats from cloned RFID

tags, malicious snooping RFID readers, and unauthorized tag's data access and manipulation can only be prevented by incorporating a tag-reader mutual authentication scheme.

The RFID tags can be made tamperproof, to make sure that the act of snatching a tag from a genuine product (pallet, carton, case, or an item) should cause a considerable damage to the tag itself rendering it permanently unusable when attached to a counterfeit product. The manufacturer of the products can also make sure that tamperproof tags are attached to the products in such a way that the act of snatching a tag causes a significant and noticeable damage to the product packaging itself. This prevents anyone trying to open a genuine package to fill inside counterfeit contents (*e.g.,* pills).

But a malicious RFID reader can easily scan and copy the data (*e.g.,* EPC number) of a genuine tag and embed the same data onto a fake tag, and this fake tag can be attached to a counterfeit product. In this case, there could be more than one product whose tags give out the same EPC number and we cannot distinguish between a genuine tag and a fake tag. And nowadays illegal marketers have become experts in producing counterfeit products that look exactly like the genuine ones. Therefore, this threat cannot be prevented by using either tamperproof tag or tamperproof packaging. Even though a particular tag gives out a genuine EPC number, it must still be authenticated by the reader in order to prove beyond doubt that the tag attached to a product is indeed from the manufacturer of the product.

Mutual authentication between RFID Tag and RFID Reader plays a vital role in identifying cloned tags, and malicious readers and thus help preventing tag and reader impersonation, unauthorized access / manipulation of tag's data by a malicious reader. The current ratified standard on EPCglobal Class 1 Gen 2 UHF RFID Protocol [4] describes only one-way reader to tag authentication by using tag's Access Password. However we show that this one-way authentication scheme is not secure and also the EPCglobal Architecture Framework specification [3] does not provide any details on how to distribute the tag's access password to the reader. Therefore in this paper we propose a simple, secure and light-weight tag - reader mutual authentication and tag's access password distribution scheme. Our scheme allows the manufacturer of the product to play a vital role in the tag-reader mutual authentication process. Therefore while authenticating a tag, the manufacturer can also implicitly keep track on the whereabouts of its products, which have genuine tags attached to them.

Ari Juels [6]summarized many previously proposed security models for tag-reader mutual authentication, but unlike these models, the main advantage of our proposed scheme is that it does not require implementation of any special cryptographic functions/keys within the tag. There is also no need for the tag and the reader to synchronize security keys/hash values. We in fact propose to improve the existing one-way reader to tag authentication scheme (described by EPCglobal Class 1 Gen 2 UHF RFID Protocol [4]) in order to accommodate tag-reader mutual authentication, using the tag's (already existing) EXOR function and access password.

Section 2 describes the RFID-based supply chain management system that adheres to EPCglobal Architecture Framework specification [3]. We explain the various entities of this framework that are related to this paper. In section 3 we introduce the one-way reader to tag authentication scheme proposed by EPCglobal Class 1 Gen 2 UHF RFID Protocol [4] and describe its security weakness. Section 4 describes our proposed tag-reader mutual authentication scheme and its analysis. Section 5 provides conclusion and our future work.

II. EPCglobal Architecture Framework

Throughout this paper we consider "EPCglobal Architecture Framework" [3] to be the typical RFID-based supply chain management system, which most of the EPCglobal Subscribers would deploy in their organization. An end-user EPCglobal Subscriber is any organization that employs EPCglobal Standards, Interfaces and EPCglobal Core Services as a part of its supply chain management system. The EPCglobal Architecture Framework specification [3] [7] provides a detailed description of the entities and their interfaces; we summarize details about some of the entities that are relevant to this paper as below:

- **RFID Tag:** Every RFID tag contains its unique Electronic Product Code (EPC) number. EPC is a globally unique serial number that identifies an item in the supply chain. EPC number contains: EPC Manager number (identifies the company), Object class (product number), Serial number (specific instance of the object class being tagged, objects own unique identifier). EPCglobal allocates manufacturers specific blocks of EPC numbers, and manufacturers then add their own product codes and serial numbers to their assigned manufacturer numbers to create unique identifiers - EPCs.
- **RFID Reader:** Make multiple observations of RFID tags while they are in the read zone.
- **RFID Middleware:** RFID Middleware filters and collects raw tag reads, over time intervals delimited by events defined by the EPCIS Capturing Application (*e.g.,* tripping a motion detector). The filtered and collected tag read data from RFID Middleware to the EPCIS Capturing Application role may say "At Location L, between time T1 and T2, the following EPCs were observed,"where the list of EPCs has no duplicates and has been filtered by criteria defined by the EPCIS Capturing Application.
- **EPCIS Accessing Application:** Responsible for carrying out overall enterprise business processes, such as warehouse management, shipping and receiving, historical throughput analysis, and so forth, aided by EPC-related data. The EPCIS Accessing Application may use the Object Name Service (ONS) to locate the EPCIS service (EPCIS Accessing Application) of the EPCglobal Subscriber who is the EPC Manager of the object in question.
- **EPCIS Query Interface:** Provides means whereby an EPCIS Accessing Application can request EPCIS data from an EPCIS Repository or an EPCIS Capturing Application, and the means by which the result is returned. Provides a means for mutual authentication of the two parties. Reflects the result of authorization decisions taken by the providing party, which may include denying a request made by the requesting party, or limiting the scope of data that is delivered in response.
- **EPCIS Repository:** Records EPCIS-level events generated by one or more EPCIS Capturing Applications, and make them available for later query by EPCIS Accessing Applications.
- **ONS:** Fulfills ONS lookup requests for EPCs within the control of the enterprise that operates the Local ONS; that is, EPCs for which the enterprise is the EPC Manager.
- **Subscriber Authentication:** Authenticates the identity of an EPCglobal Subscriber. Provides credentials that one EPCglobal Subscriber may use to authenticate itself to another EPCglobal Subscriber, without prior arrangement between the two Subscribers. Authenticates participation in network services through validation of active EPCglobal Subscription.

III. Security Assessment of Class 1 Gen 2 UHF RFID Protocol

1. Overview

A tag can be embedded with a 32-bit value Access Password, which means that only a reader that already possesses the right access password can perform mandatory commands on the tag, such as Read, Write, and Lock. Therefore tag's access password can be used for "reader to tag" authentication and in the process allows the reader to access the locked memory banks within the tag, permission to change the lock status of the memory banks, and write data into the tag, etc.

A tag has four memory banks: Reserved, EPC, TID, and User. Reserved memory bank is used to store the Kill Password & Access Password, EPC memory bank for EPC number, TID memory bank for tag's unique manufacturer identity number, and User memory bank for additional user data. The current standard allows the tag to reveal only its EPC number, but rest of its memory banks (TID, User) can be locked and be accessed if and only if the reader presents the right access password. The reserved memory bank of the tag is permanently locked; as a result the access password can neither be read nor modified by any reader.

Tags shall implement a random or pseudo-random number generator (RNG). Tags generate 16-bit random or pseudo-random numbers (RN16) using the RNG. Tags have the ability to temporarily store at least two RN16s while powered.

- Probability of a single RN16: The probability that any RN16 drawn from the RNG has value RN16=j, for any j, shall be bounded by $0.8/216<P(RN16=j)<1.25/216$.
- Probability of simultaneously identical sequences: For a tag population of up to 10,000 tags, the probability that any two or more Tags simultaneously generate the same sequence of RN16s shall be less than 0.1%, regardless of when the Tags are energized.
- Probability of predicting an RN16: An RN16 drawn from a tag's RNG 10ms after the end of RF signal envelope rise time shall not be predictable with a probability greater than 0.025% if the outcomes of prior draws from the RNG, performed under identical conditions, are known.

Readers and tags may implement an Access command; Access causes a tag with an access password to transition from the open to the secured state. Reader and tag can communicate indefinitely in the secured state. Just prior to issuing each Access command the reader first issues a Req_RN to obtain a new RN16. RN16s are used has EXOR pads to obscure access password being sent from the reader to tag. To access a tag, a reader shall follow a multi-step procedure as shown in Fig. 1, where a reader issues two Access commands, the first containing the 16 MSBs (Most Significant Bits) of the tag's access password (PWD_M) EXORed (Exclusive Disjunction) with an RN16_1, and the second containing the 16 LSBs (Least Significant Bits) of the tag's access password (PWD_L) EXORed with a different RN16_2. Each EXOR operation shall be performed MSB first (i.e. the MSB of each half-password shall be EXORed with the MSB of its respective RN16). This method of obscuring the 16-bit access password chunks EXORed with RN16s is known as Cover-Coding. The tag receives both the cover-coded password chunks (CCPWD_M and CCPWD_L) and EXORs CCPWD_M with RN16_1 and CCPWD_L with RN16_2 to verify whether the results equal to PWD_M and PWD_L respectively. If yes the tag is sure that the reader is authentic, if not the tag ends the communication channel with the reader.

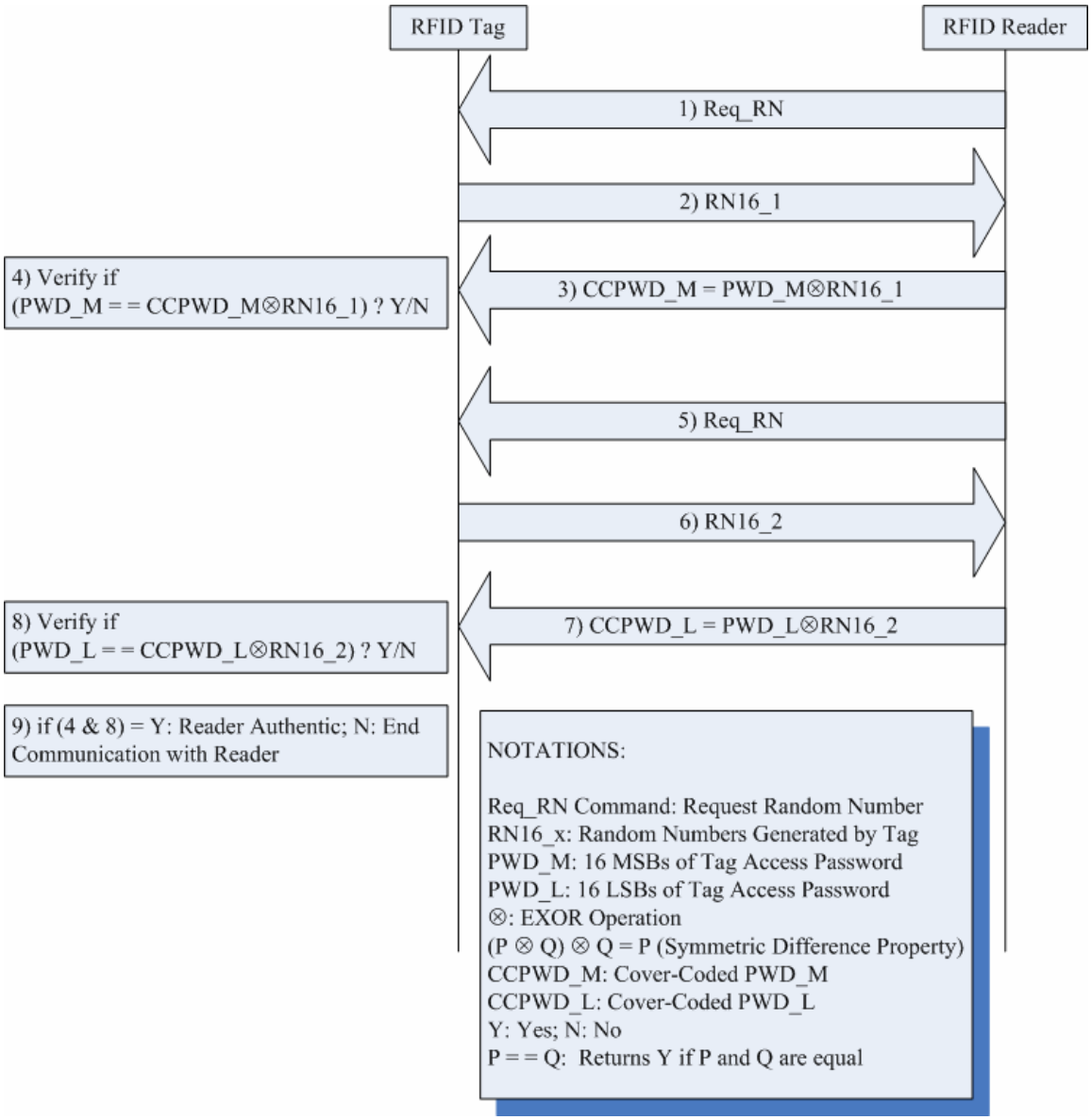

Figure 1: One-Way Reader to Tag Authentication Proposed by Class 1 Gen 2 UHF RFID Protocol Standard

2. Analysis

- **Access Password Exposed:** The above-mentioned approach is not all secure because the tag sends both the RN16s in unencrypted form. Therefore any eavesdropping or a disgruntled/compromised employee holding a handheld reader can easily read/capture these RN16s, and due to symmetric difference property of EXOR: (RN16 EXOR AccessPwd) EXOR RN16 = AccessPwd, the access password can be easily exposed.
- **Tag Cloning:** The access password is embedded within the tag by the

manufacturer of the product, and this access password remains the same for the rest of the product's life cycle. Therefore, an exposed access password at any of the EPCglobal Subscribers end would easily assist an adversary to create cloned fake tags with the same access password. These fake tags can later be attached to counterfeit products, which can be introduced into the supply chain.

- **Tag's Data Manipulation:** Once the access password is exposed, any malicious reader can access the locked memory banks of that particular tag and either corrupt or manipulate its data.

IV. Proposed Tag-Reader Mutual Authentication Scheme

In this section we describe our proposed access password based tag-reader mutual authentication, tag access password distribution scheme, and also simultaneous update of manufacturer's EPCIS Repository with the arrival information of a particular consignment at any of the other EPCglobal Subscriber's end.

1. Scheme

Considering the supply chain scenario mentioned in Section 2.1, let us assume that a distributor receives a pallet of a particular product from a manufacturer. The distributor wants to authenticate the tag attached to the pallet in order to make sure that the pallet has indeed come from the manufacturer, and the contents of the pallet are intact. But the distributor does not know the tag's access password. The distributor's EPCIS Accessing Application (EAA-D) contacts the manufacturer's EPCIS Accessing Application (EAA-M) in order to get the access password of the tag attached to the pallet. Since giving away the access password to the distributor would compromise the security of the tag for the rest of the supply chain, the EAA-M, EAA-D, reader and the tag follow a multi-step procedure (Fig. 2) described in this section to accomplish tag-reader mutual authenticate. EPCglobal Subscribers can follow the similar procedure to authenticate the tags attached to the cartons, cases, and individual items. For reasons of clarity, the CRC-16 (Cyclic Redundancy Check) and Handle (tag-reader communication session identifier) bits have been omitted from all commands and replies between the tag and the reader.

The main part of our proposed scheme is "pad generation phase". In this phase two 16-bit random numbers from both the tag and the EAA-Mare used to generate two 16-bit pads. These pads are generated using the access password, and only the tag and the EAA-M have the knowledge of the access password. Therefore just by sharing the random numbers among them, both the tag and EAA-Mcan generate the same pads internally. Later these pads are used to cover-code the two 16-bit access password chunks to mutually authenticate each other. This approach prevents the major drawback of the one-way authentication scheme proposed by EPCglobal Class 1 Gen 2 UHF RFID Protocol [4], where random numbers sent in unencrypted form are used as pads to cover-code the access password chunks. But in our proposed scheme the generated pads are known only to the tag and the EAA-M and using them to cover-code the access password chunks provides good obscurity and security to the real access password. The reader, EAA-D, and the other EPCglobal Subscribers (apart from the manufacturer of the product) receive only the cover-coded access password chunks from the EAA-M, and only the tag in question has the right pad to verify the cover-coded access password chunks from the EAA-M and similarly only the EAA-M has the right pad to verify the cover-coded access password chunks from the tag (mutual authentication). Therefore we can fend off threats like exposed tag's access password, malicious snooping readers, disgruntled EPCglobal Subscriber employee managing hand-held reader, and cloned tags

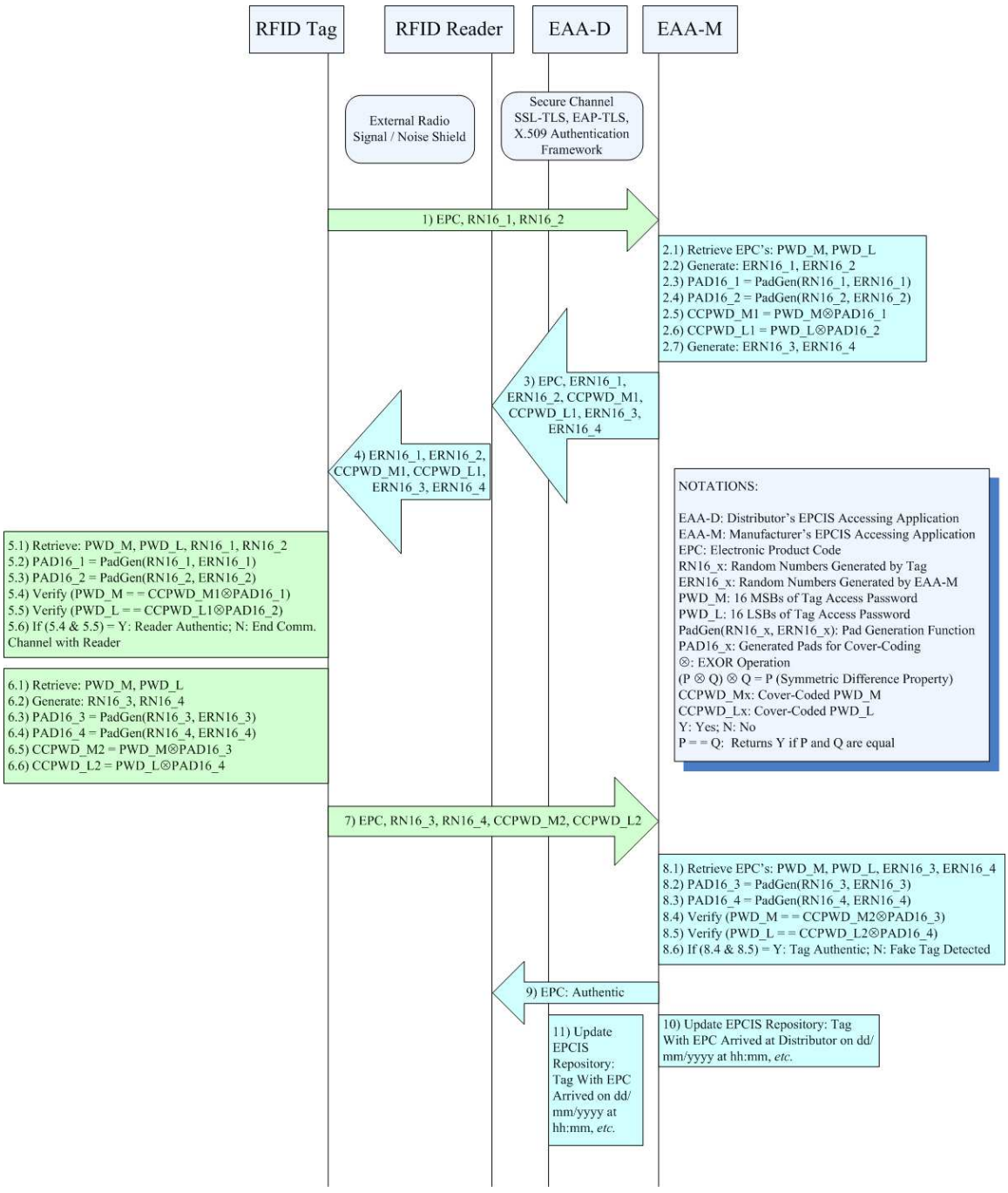

Figure 2: Proposed Tag-Reader Mutual Authentication Scheme

In this scheme we consider four entities: RFID Tag, RFID Reader, EAA-D, and EAA-M. For reasons of clarity we push the roles of RFID Middleware, and EPCIS Repository into the background; however it should be noted that these two entities still play a vital role

in the communication chain. We assume that the communication channel between the resource rich entities like RFID Reader, EAA-D, and EAA-M to be highly secure with the use of technologies like SSL-TLS (wired channel), EAP-TLS (wireless Channel), and X.509 Authentication Framework. The trusted "Subscriber Authentication"core service identifies the roles (distributor, wholesaler, and retailer) of various EPCglobal Subscribers and distributes appropriate X.509 certificates to them. These certificates authenticate and authorize the subscribers to each other and help in securing the communication channel among them. Whereas the communication RF signals between the RFID tag and RFID reader can be partially secured by installing external radio signal/noise shield.

Step 1:

- The tag has been singulated by the reader and is in the acknowledged state. Tag backscatters its EPC number.
- Reader sends command Req_RN. Tag backscatters two generated random numbers: RN16_1 and RN16_2 and stores them in its memory (for later use in Steps 5.1, 5.2, & 5.3). Details about random generator and random numbers are provided in the subsection 3.1 For example,

  o RN16_1 = A69Dh

  o RN16_2 = 7E2Bh

(h represents hexadecimal)

- Since the reader has no clue about the access password of the tag, the reader sends (EPC, RN16_1, RN16_2) to the EAA-D (via the RFID Middleware and EPCIS Repository). EAA-D sends (EPC, RN16_1, RN16_2) to EAA-M.
- EAA-D and EAA-M authenticate each other based on the X.509 Authentication Framework and setup secure communication tunnel based on either SSL-TLS (wired channel) and EAP-TLS (wireless channel). "Subscriber Authentication"core service provides the needed X.509 digital certificates and public-private secret keys to EAA-D and EAA-M. If the authentication between the EAA-D and EAA-M fails, then EAA-M would not accept any data from EAA-D.

Step 2:

Step 2.1: EAA-M uses EPC to query the manufacturer's EPCIS Repository and retrieves the corresponding access password.

- For example, let us assume:

  o The Tag's access password is AC9EC5D6h.

  o The 1st half (16 MSBs) of the access password is PWD_M = AC9Eh.= 1010 1100 1001 1110

  o The 2nd half (16 LSBs) of the access password is PWD_L = C5D6h = 1100 0101 1101 0110

Step 2.2: EAA-M generates two 16-bit value random numbers ERN16_1 and ERN16_2

- For example,

  o ERN16_1 = 2B5Fh

  o ERN16_2 = D487h

Step 2.3 (Pad Generation Phase): PadGen(RN16_1, ERN16_1): As per the EPCglobal Class 1 Gen 2 UHF RFID Protocol [4]ratified standard, the access password is a 32-bit value stored in tag's Reserved memory 20h to 3Fh, MSB first. Figure 3 depicts tag's logical memory and access password map, including the bit-wise storage of our example access password.

EAA-M uses the value of RN16_1 (A69Dh) and ERN16_1 (2B5Fh) as location (Locn.) numbers (ref. Fig. 3) to retrieve the individual access password bits stored in those locations, concatenate these bits in series to form a 16-bit value PAD16_1.

- For example: PadGen(A69Dh , 2B5Fh)

  o A69Dh = 10th 6th 9th 13th location of Access Password's MSBs (20h to 2Fh)= 0001

  o A69Dh = 10th 6th 9th 13th location of Access Password's LSBs (30h to 3Fh)= 0011

  o 2B5Fh = 2nd 11th 5th 15th location of Access Password's MSBs (20h to 2Fh)= 1110

  o 2B5Fh = 2nd 11th 5th 15th location of Access Password's LSBs (30h to 3Fh)= 0110

  o Combining the above 4 results we have a 16-bit pad valuePAD16_1 = 0001 0011 1110 0110 = 13E6h

Step 2.4 (Pad Generation Phase): PadGen(RN16_2, ERN16_2): Similarly as described above EAA-M uses the value of RN16_2 (7E2B h) and ERN16_2 (D487 h) as location (Locn.) numbers (ref. Fig. 3) to retrieve the individual access password bits stored in those locations, concatenate these bits in series to form a 16-bit value PAD16_2.

- For example: PadGen(7E2Bh , D487h)

  o 7E2Bh = 7th 14th 2nd 11th location of Access Password's MSBs (20h to 2Fh)= 0111

  o 7E2Bh = 7th 14th 2nd 11th location of Access Password's LSBs (30h to 3Fh)= 1101

  o D487h = 13th 4th 8th 7th location of Access Password's MSBs (20h to 2Fh)= 1110

  o D487h = 13th 4th 8th 7th location of Access Password's LSBs (30h to 3Fh)= 1011

  o Combining the above 4 results we have a 16-bit pad valuePAD16_2 = 0111 1101 1110 1011 = 7DEBh

Steps 2.5 & 2.6: EAA-M performs the following bit-wise EXOR () operations between the two 16-bit access password chunks and the generated pads: (PWD_M PAD16_1) and (PWD_L PAD16_2) to obtain cover-coded access password (16-bit) chunks: CCPWD_M1 and CCPWD_L1 respectively.

- For example,

  o CCPWD_M1 = PWD_M ⊗ PAD16_1 = AC9Eh ⊗ 13E6h = BF78h

  o CCPWD_L1 =PWD_L ⊗ PAD16_2 = C5D6h ⊗ 7DEBh = B83Dh

Step 2.7: EAA-M generates two more 16-bit value random numbers ERN16_3 and ERN16_4 and stores them (for later use in Steps 8.2 & 8.3).

- For example,

  o ERN16_3 = 62FBh

  o ERN16_4 = 5AD3h

Step 3:

- EAA-M sends (EPC, ERN16_1, ERN16_2, CCPWD_M1, CCPWD_L1, ERN16_3, ERN16_4) to EAA-D.

- EAA-D and EAA-M authenticate each other based on the X.509 Authentication Framework and setup secure communication tunnel based on either SSL-TLS (wired channel) and EAP-TLS (wireless channel). "Subscriber Authentication"core service provides the needed X.509 digital certificates and public-private secret keys to EAA-D and EAA-M. If the authentication between the EAA-D and EAA-M fails, then EAA-D would not accept any data from EAA-M.

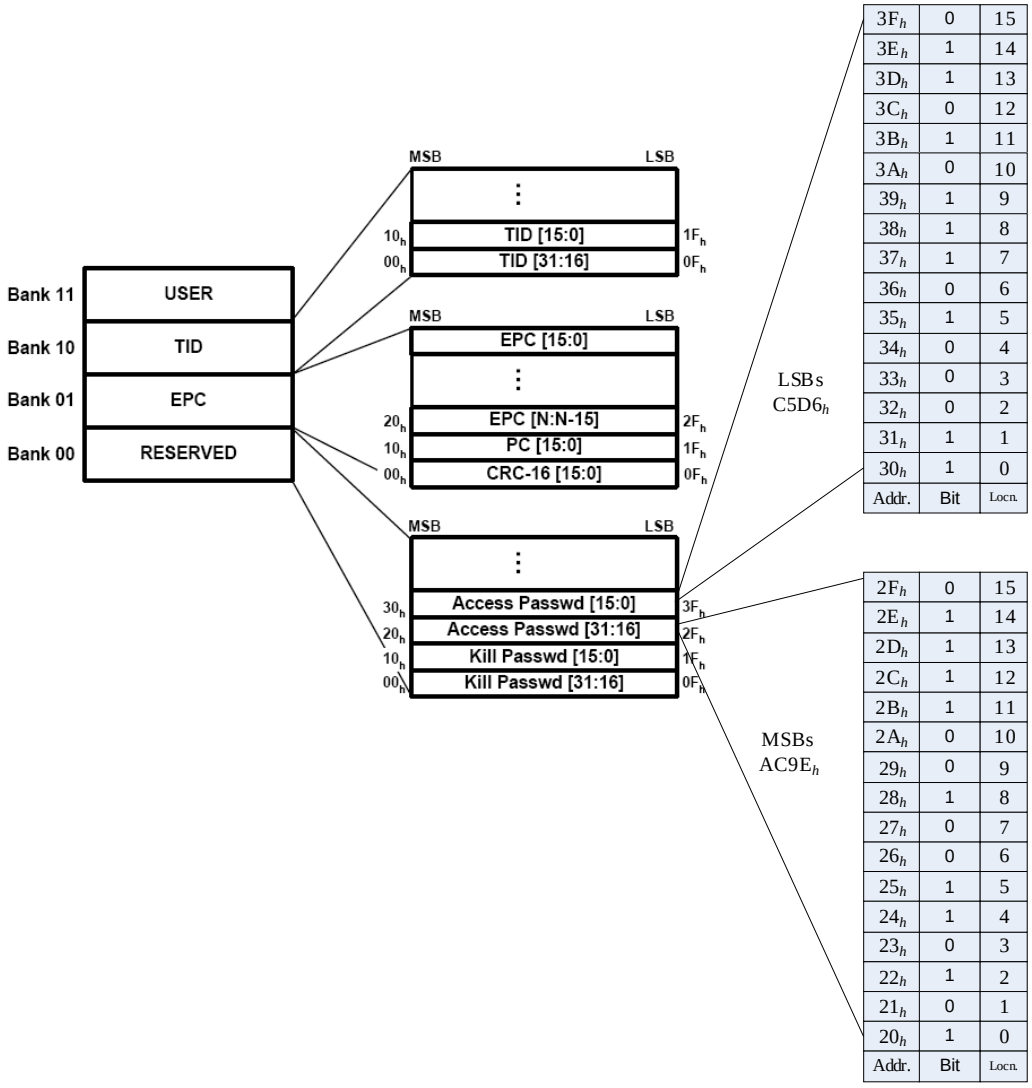

Figure 3: Tag's Logical Memory & Access Password Map

Step 4:

EAA-D sends (ERN16_1, ERN16_2, CCPWD_M1, CCPWD_L1, ERN16_3, ERN16_4) to the reader. From the reader the tag receives (ERN16_1, ERN16_2, CCPWD_M1, CCPWD_L1, ERN16_3, ERN16_4)

Step 5:

Steps 5.1, 5.2 & 5.3: Tag received ERN16_1, and ERN16_2 from EAA-M (Step 4). The tag already hasthe two 16-bit chunks of the access password (PWD_M and PWD_L) embedded within its Reserved memory, and the initially generated two random numbers: RN16_1 and RN16_2(Step 1). This information is sufficient for the tag to perform the same pad generation phase mentioned in Steps 2.3

& 2.4, and generate the same pads: PAD16_1 and PAD16_2 from the access password.

- For example: PadGen(RN16_1, ERN16_1) & PadGen(RN16_2, ERN16_2)
- For example: PadGen(A69Dh , 2B5Fh) & PadGen(7E2B h , D487h)

o PAD16_1 = 0001 0011 1110 0110 = 13E6h

o PAD16_2 = 0111 1101 1110 1011 = 7DEBh

Steps 5.4, 5.5, & 5.6: Having generated the pads: PAD16_1 and PAD16_2, the tag uses the cover-coded chunks (CCPWD_M1, CCPWD_L1) received from EAA-M in Step 4 to verify if:

- PWD_M == CCPWD_M1 ⊗ PAD16_1 ? Y: Reader Authentic; N: End Communication Channel with Reader
- PWD_L == CCPWD_L1 ⊗ PAD16_2 ? Y: Reader Authentic; N: End Communication Channel with Reader

o BF78 h 13E6h = 1011 1111 0111 10002 ⊗ 0001 0011 1110 0110 = AC9Eh = PWD_M

o B83D h 7DEB h = 1011 1000 0011 11012 ⊗ 0111 1101 1110 1011 = C5D6h = PWD_L

If the verification is successful then the tag would authenticate the reader to be a genuine entity of the EPCglobal Subscriber (distributor) trusted by the manufacturer of the product. Otherwise the tag ends the communication channel with the reader and returns to arbitrate state, in which case the malicious reader trying to gain illegal access to the tag can neither perform commands like Read, Write, Lock, etc. nor can get the real access password.

Step 6:

It's the tag's turn to perform similar steps from Step 2.2 through Step 2.6 and finally return two new cover-coded access password (16-bit) chunks: CCPWD_M2 and CCPWD_L2 respectively in order to authenticate itself to the EAA-M, EAA-D, and the reader.

Step 6.1: The tag fetches its access password from the Reserved Memory. As mentioned above, in our example, the Tag's access password is AC9EC5D6h.

- The 1st half (16 MSBs) of the access password isPWD_M = AC9Eh.= 1010 1100 1001 1110
- The 2nd half (16 LSBs) of the access password is PWD_L = C5D6h = 1100 0101 1101 0110

Step 6.2: Tag generates two 16-bit value random numbers RN16_3 and RN16_4

- For example,

o RN16_3 = 1A5Eh

o RN16_4 = C38Fh

Step 6.3: (Pad Generation Phase): PadGen(RN16_3, ERN16_3): Similar to Step 2.3. Tag already received two extra random numbers: ERN16_3, and ERN16_4, generated by the EAA-M during the Step 4.

o ERN16_3 = 62FBh

o ERN16_4 = 5AD3h

Tag uses the value of RN16_3 (1A5Eh) and ERN16_3 (62FBh) as location (Locn.) numbers (ref. Fig. 3) to retrieve the individual access password bits stored in those locations, concatenate these bits in series to form a 16-bit value PAD16_3.

- For example: PadGen(1A5Eh , 62FBh)

o 1A5Eh = 1st 10th 5th 14th location of Access Password's MSBs (20h to 2Fh)= 0011

o 1A5Eh = 1st 10th 5th 14th location of Access Password's LSBs (30h to 3Fh)= 1010

o 62FBh = 6th 2nd 15th 11th location of Access Password's MSBs (20h to 2Fh)= 0101

o 62FBh = 6th 2nd 15th 11th location of Access Password's LSBs (30h to 3Fh)= 0001

o Combining the above 4 results we have a 16-bit pad valuePAD16_3 = 0011 1010 0101 0001 = 3A51h

Step 6.4: (Pad Generation Phase): PadGen(RN16_4, ERN16_4): Similar to Step 2.4. Similarly as described above tag uses the value of RN16_4 (C38F h) and ERN16_4 (5AD3h) as location (Locn.) numbers (ref. Fig. 3) to retrieve the individual access password bits stored in those locations, concatenate these bits in series to form a 16-bit value PAD16_4.

- For example, PadGen(C38F h , 5AD3h)

o C38Fh = 12th 3rd 8th 15th location of Access Password's MSBs (20h to 2Fh)= 1010

o C38Fh = 12th 3rd 8th 15th location of Access Password's LSBs (30h to 3Fh)= 0010

o 5AD3h = 5th 10th 13th 3rd location of Access Password's MSBs (20h to 2Fh)= 1010

o 5AD3h = 5th 10th 13th 3rd location of Access Password's LSBs (30h to 3Fh)= 1010

o Combining the above 4 results we have a 16-bit pad valuePAD16_4 = 1010 0010 1010 1010 = A2AAh

Step 6.5 & 6.6: The tag performs the following bit-wise EXOR () operations between the two 16-bit access password chunks and the generated pads: (PWD_M PAD16_3) and (PWD_L PAD16_4)to obtain cover-coded access password (16-bit) chunks: CCPWD_M2 and CCPWD_L2 respectively.

- For example,

o CCPWD_M2 = PWD_M ⊗ PAD16_3 = AC9E h ⊗ 3A51h = 96CFh

o CCPWD_L2 =PWD_L ⊗ PAD16_4 = C5D6 h ⊗ A2AAh = 677Ch

Step 7:

- The tag sends (EPC, CCPWD_M2, CCPWD_L2, RN16_3, RN16_4) to the reader. The reader sends (EPC, CCPWD_M2, CCPWD_L2, RN16_3, RN16_4) to the EAA-D. The EAA-D sends (EPC, CCPWD_M2, CCPWD_L2, RN16_3, RN16_4) to EAA-M for verification and confirmation that the tag is authentic/genuine.

- EAA-D and EAA-M authenticate each other based on the X.509 Authentication Framework and setup secure communication tunnel based on either SSL-TLS (wired channel) and EAP-TLS (wireless channel). "Subscriber Authentication" core service provides the needed X.509 digital certificates and public-private secret keys to EAA-D and EAA-M. If the authentication between the EAA-D and EAA-M fails, then EAA-M would not accept any data from EAA-D.

Step 8:

Step 8.1: EAA-M uses EPCto query the EPCIS Repository of the manufacturer and retrieves the corresponding access password. As mentioned above, in our example, the Tag's access password is AC9EC5D6h.

- The 1st half (16 MSBs) of the access password is PWD_M = AC9E h.= 1010 1100 1001 1110

- The 2nd half (16 LSBs) of the access password is PWD_L = C5D6 h = 1100 0101 1101 0110

Step 8.2 & 8.3: EAA-M received RN16_3, and RN16_4 from EAA-D in Step 7. EAA-M already has the two 16-bit chunks of the access password (PWD_M and PWD_L) stored in its EPCIS Repository, and the two random numbers: ERN16_3 and ERN16_4 generated in Step 2.7. This information is sufficient for EAA-M to perform same pad generation phase mentioned in Steps 6.3 & 6.4, and generate the same pads: PAD16_3 and PAD16_4 from the access password.

- For example: PadGen(RN16_3, ERN16_3) & PadGen(RN16_4, ERN16_4)
- For example: PadGen(1A5Eh , 62FBh) & PadGen(C38F h , 5AD3h)

o PAD16_3 = 0011 1010 0101 0001 = 3A51h

o PAD16_4 = 1010 0010 1010 1010 = A2AAh

Steps 8.4, 8.5, & 8.6: Having generated the pads: PAD16_3 and PAD16_4, the EAA-M uses the cover-coded chunks (CCPWD_M2, CCPWD_L2) received from the EAA-D in Step 7 to verify if:

- PWD_M == CCPWD_M2 ⊗ PAD16_3 ? Y: Tag Authentic; N: Fake Tag Detected
- PWD_L == CCPWD_L2 ⊗ PAD16_4 ? Y: Tag Authentic; N: Fake Tag Detected

o 96CFh ⊗ 3A51h = 1001 0110 1100 11112 ⊗ 0011 1010 0101 0001 = AC9Eh = PWD_M

o 677Ch ⊗ A2AAh = 0110 0111 0111 11002 ⊗ 1010 0010 1010 1010 = C5D6h = PWD_L

If the verification is successful then the EAA-M would authenticate the tag to be genuine. Otherwise the EAA-M would raise an alarm indicating a fake product in the supply chain.

Step 9, 10 & 11:

- EAA-M after authenticating the tag in Steps 8.4, 8.5, & 8.6, informs EAA-D that the tag in question is indeed genuine. EAA-D passes on this information to the EPCIS Repository and then to the RFID Middleware and finally to the reader. EAA-M also updates the EPCIS Repository of the manufacturer with information that the container to which the tag is attached to has reached the distributor and other information associated with this event.
- EAA-D and EAA-M authenticate each other based on the X.509 Authentication Framework and setup secure communication tunnel based on either SSL-TLS (wired channel) and EAP-TLS (wireless channel). "Subscriber Authentication" core service provides the needed X.509 digital certificates and public-private secret keys to EAA-D and EAA-M. If the authentication between the EAA-D and EAA-M fails, then EAA-D would not accept any data from EAA-M.

2. Analysis

1) Tag's Access Password Never Exposed:

Our scheme does not use the random numbers sent in unencrypted form as pads to cover-code the tag's access password. Instead these random numbers are used in association with the tag's access password to generate pads. As different random numbers are used during different sessions, even the pads are different. These generated pads are known only to the tag and the EAA-M and using these pads to cover-code the two 16-bit access password chunks provides good obscurity and security to the real tag's access password. The reader, EAA-D, and the other EPCglobal Subscribers (apart from the manufacturer of the product) receive only the cover-coded access password chunks from the EAA-M, and only the tag in question has the right pad to verify the cover-coded access password chunks from the EAA-M and similarly only the EAA-M has the right pad to verify the cover-coded access password chunks from the tag (mutual authentication). Therefore the access password is never exposed, even to any of the other EPCglobal Subscribers, thus protecting the tag's access password which plays a vital role in authenticating the tag thorough out the supply chain. We can fend off threats from malicious snooping readers, disgruntled EPCglobal Subscriber employee managing hand-held reader, and cloned tags with genuine access passwords embedded within them.

2) Tag-Reader Mutual Authentication:

The generated pads are known only to the tag and the EAA-M and these pads are used to cover-code the two 16-bit access password chunks. EAA-M and EAA-D authenticate each other and establish a secure communication channel (SSL-TLS, EAP-TLS, X.509 Authentication Framework). EAA-M sends the two cover-coded 16-bit access password chunks to the EAA-D, which in turn sends the chunks to the reader. EAA-D and the reader must have already authenticated and established a secure communication channel. Thus the tag receives the two cover-coded 16-bit access password chunks from the reader and if the verification of the cover-coded chunks is successful, the tag authenticates the reader to be genuine and that the reader is a part of the EPCglobal Subscriber (distributor) trusted and authenticated by the manufacturer of the product. We can fend off threats like unauthorized tag's data access and manipulation by a malicious reader in the vicinity of the supply chain processing.

3) Secure Against Replay Attacks:

We use random numbers generated by both the tag and the EAA-M to generate each of the pads. Therefore replaying a particular session would not serve any purpose. Also for additional security, EAA-M or the RFID Middleware at the distributor end must detect weak random numbers (*e.g.,* 0000h, 1111h, FFFFh, *etc.*) from the tag that can compromise tag's access password.

4) Light-Weight Mutual Authentication:

Our scheme does not use any special cryptographic functions. The tag already has the capability to compute EXOR operations, temporarily store random numbers and fetch the access password embedded within its Reserved Memory. Therefore our scheme utilizes the features described in EPCglobal Class 1 Gen 2 UHF RFID Protocol [4]ratified standard. Our scheme requires the tag to temporarily store five random numbers (handle, two RN16-x, and two ERN16_x)when compared to two random numbers for the one-way reader to tag authentication scheme proposed by EPCglobal Class 1 Gen 2 UHF RFID Protocol [4]ratified standard. We also require the tag to perform a very simple pad generation function which collects the individual bits of the access password from the memory locations identified by these random numbers. Therefore our proposed scheme requires very minute changes to the EPCglobal Class 1 Gen 2 UHF RFID Protocol [4] ratified standard.

5) Access Password Scalability:

EPCglobal Class 1 Gen 2 UHF RFID Protocol [4] ratified standard has proposed only a 32-bit access password to be embedded within the tag. But a 32-bit password is not very secure against active attacks and fails when brute-force attack is mounted. We did not want to cause a major change in the ratified standard so we adhered to the 32-bit access password and further enhanced its security with very minor tweaks. Our light-weight scheme can still be applicable in case the length of the access password is extended.

V. Conclusion and Future Work

The threats from cloned fake RFID tags and malicious RFID readers can only be prevented by incorporating a tag-reader mutual authentication scheme. In this paper we proposed a simple, secure, and light-weight tag-reader mutual authentication scheme that adheres to both EPCglobal Architecture Framework specification and EPCglobal Class1 Gen 2 UHF RFID Protocol ratified standard. This scheme emphasizes on the importance of tag's access password, which can play a vital role in tag-reader mutual authentication. Tag's access password is never exposed throughout the supply chain, even to the other EPCglobal Subscribers and is only known to the genuine tag and the manufacturer of the product or the subscriber who owns the tag. Our

scheme allows the manufacturer of the product to play an active role in the mutual authentication procedure and therefore a manufacturer can immediately know that a particular genuine tag attached to a product (container, pallet, carton, case, and item) has reached the other EPCglobal Subscriber's end. Finally, our scheme also describes the method to distribute tag's access password to only genuine RFID readers.

Our "Pad Generation Phase"approach prevents the major drawback of the one-way authentication scheme proposed by EPCglobal Class 1 Gen 2 UHF RFID Protocol [4], where random numbers sent in unencrypted form are used as pads to cover-code the access password chunks. But in our proposed scheme the generated pads are known only to the tag and the manufacturer of the product and using them to cover-code the access password chunks provides good obscurity and security to the real access password., the main advantage of our proposed scheme is that it does not require implementation of any special cryptographic functions/keys within the tag. There is also no need for the tag and the reader to synchronize security keys/hash values. Our scheme would improve the existing one-way reader to tag authentication scheme (described by EPCglobal Class 1 Gen 2 UHF RFID Protocol)in order to accommodate tag-reader mutual authentication, using the tag's (already existing) EXOR function and access password.

Our future work includes developing a secure approach where the consumer can use his/her mobile phone (Mobile RFID technology) to scan a particular tag attached to an item and then connect to the manufacturer's EPCIS Accessing Application in order to verify if the product is genuine or fake. In this way consumers can participate in the fight against counterfeit products by easily detecting and notifying authorities if they come across any counterfeit products.

References

[1] EPCglobal IncTM, http://www.epcglobalinc.org/

[2] VeriSign (2005), "The EPCglobal Network: Enhancing the Supply Chain", White Paper 2005,http://www.verisign.com/stellent/groups/public/documents/white_paper/002109.pdf

[3] EPCglobal Architecture Framework Version 1.0 (2005), EPCglobal Specification, http://www.epcglobalinc.org/standards/

[4] EPCTM Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz 960MHz Version 1.0.9 (2005), EPCglobal Ratified Standard, http://www.epcglobalinc.org/standards/

[5] EPCglobal Certificate Profile (2006), EPCglobal Ratified Standard, http://www.epcglobalinc.org/standards/

[6] Ari Juels(2005), "RFID Security and Privacy: A Research Survey", RSA Laboratories.

[7] Divyan M. Konidala, Woan-Sik Kim, Kwangjo Kim (2006), "Security Assessment of EPCglobal Architecture Framework", Whitepaper series on Anti-counterfeiting and Secure Supply Chain, Auto-ID Labs.