A Simple and Cost-effective RFID Tag-Reader Mutual Authentication Scheme

Divyan M. Konidala, Zeen Kim, Kwangjo Kim

Information and Communications University (ICU),
International Research Center for Information Security (IRIS), Auto-ID Lab Korea
R504, 103-6, MunjiDong, Daejeon 305732, Republic of Korea
{divyan, zeenkim, kkj}@icu.ac.kr

Abstract. Cloned fake RFID tags and malicious RFID readers pose a major threat to RFID-based supply chain management system. Fake tags can be attached to counterfeit products and medicines. Malicious readers can corrupt and snoop on genuine tags. These threats can be alleviated by incorporating a RFID tag-reader mutual authentication scheme. In this paper we propose a simple, cost-effective, light-weight, and practical RFID tag-reader mutual authentication scheme. Our scheme adheres to two ratified standards: EPCglobal Architecture Framework specification and EPCglobal Class 1 Gen 2 UHF RFID Protocol. This scheme utilizes the tag's Access and Kill Passwords and achieves the following three goals: detect cloned fake tags, ward off malicious snooping readers, and in the process, a manufacturer can also implicitly keep track on the whereabouts of its genuine products.

1 Introduction

1.1 RFID Technology

Radio Frequency IDentification (RFID) technology offers strategic advantages for businesses because it can provide efficient real-time product track and trace capability. VeriSign [1] gives a detailed description about advantages of RFID technology for supply chain management. With RFID technology, manufacturers attach Passive-RFID tags to their products. Most of these tags contain only a unique Electronic Product Code (EPC) number and further information about the product (e.g., product description, manufacturing date, packaging, shipments, product arrival and departure details, *etc.*) is stored on a network of databases, called the EPC-Information Services (EPC- IS). A RFID reader uses EPC number to locate the right EPC-IS, from where it can download and upload data about the product it scanned. Therefore, EPC-IS assists geographically distributed supply chain stakeholders to easily and efficiently access and share information on any product they are handling. EPCglobal Inc [2] is leading the development of industry-driven standards for the EPC to support the use of RFID in supply chain management. We composed this paper based on the following ratified standards: (i) EPCglobal Architecture Framework [3], (ii) EPCglobal Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz 960MHz [4], (iii) EPCglobal Certificate Profile [5].

1.2 Security Threats and Requirements

In this paper we identified and focused on the following security threats and security requirements.

Threat 1: RFID Tag Snatching: RFID tags can be made tamperproof, so that snatching a tag from a genuine product (pallet, case, or an item) should render itself permanently unusable to be re-attached to a counterfeit product.

Threat 2: Malicious RFID Readers: A RFID tag always responds with it's EPC number to any querying RIFD reader. Therefore a powerful malicious reader can illegally snoop upon the tags (attached to products) inside a container, warehouse, *etc*, leading to corporate espionage. Such readers can also corrupt and modify the tag's data. Therefore, a tag must be able to authenticate its reader. Also, only authorized readers must be allowed to access the EPC-IS.

Threat 3: RFID Tag Cloning: A malicious reader can easily scan and copy the data (e.g., EPC number) on a genuine tag and embed the same data onto a fake tag. This fake tag can be attached to a counterfeit product. This threat cannot be prevented by tamperproof tags. Even though a particular tag gives out a genuine EPC number, it must still be authenticated by the reader.

Threat 4: Insider Attack: The current ratified standard on EPCglobal Class 1 Gen 2 UHF RFID Protocol [4] describes only a one-way reader-to-tag authentication scheme. As per this standard, the manufacturer of a product can embed a unique 32-bit *Access Password* (APwd) into the tag. Only a reader with the right APwd can communicate with the tag. This scheme is not secure and it does not provide details on the secure distribution of the tag's APwd from the manufacturer of the product to the stakeholder's (*e.g.,* distributor, retailer) RFID reader. Any disgruntled, or compromised employee, can easily obtain the APwd by eavesdropping on any one of the communication sessions between the tag and the reader. The APwd for a tag, remains the same for the rest of the product's life cycle. Therefore, an exposed APwd at any of the stockholders end, would easily lead to fabrication of cloned fake tags with the same APwd. It would also allow any malicious reader to illegally access, corrupt or manipulate tag's data. Therefore we need a two-way tag-reader mutual authentication scheme, and obscure the APwd during a communication session.

Threat 4: Man-in-the-Middle Attack: To accommodate quick and speedy scanning of goods in large bulks, EPCglobal Class 1 Gen 2 UHF RFID tags exhibit outstanding far-field performance. Readers can query and communicate with these tags over a range of 10 meters. Therefore, we can anticipate Man-in-the-Middle attacks from powerful malicious readers. This attack can be mounted to eavesdrop on the communication channel between the tag and the reader and to capture a tag's EPC number and its APwd. To alleviate this threat we need to incorporate a tag-reader mutual authentication scheme, cover-code or obscure the APwd during the communication session, and finally the supply chain processing facility must be well-shielded from malicious external RF signals/noise.

1.3 Contributions of this Paper

In order to alleviate the above mentioned threats, in this paper we propose the following:

- Cheap passive-tags have tightly constrained computational and memory resources. Therefore we propose a simple, cost-effective, light-weight, and practical tag - reader mutual authentication scheme.
- A better approach to cover-code or obscure tag's *Access Password* (APwd)
- Secure distribution of obscured tags' APwd to stakeholder's RFID readers
- The manufacturer of the product plays a vital role in the tag-reader mutual authentication process. Therefore, the manufacturer can also implicitly keep track on the whereabouts of its products.
- Our scheme adheres to EPCglobal: Architecture Framework specification [3], Class 1 Gen 2 UHF RFID Protocol [4], and Certificate Profile [5]

Juels [6] summarized many previously proposed tag-reader authentication schemes. Some of the proposed solutions like [7], depend on hash function. But due to constrained resources, Class-1 Gen-2 tags are not capable of executing cryptographic hash function like MD5 and SHA-1. M^2AP [8] claims to be an ultra-lightweight RFID mutual authentication protocols, which uses only simple bitwise operations. But [9] shows that this protocol fails under Desynchronization attack, and Full-disclosure attack. Unlike these schemes, the main advantage of our proposed scheme is that it does not require the implementation of any special cryptographic hash 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 (proposed by EPCglobal) to also accommodate tag-reader mutual authentication. Our scheme utilizes tag's already existing, 16-bit random number generator, XOR function, and *Access & Kill Passwords*. Our scheme is not fully secure but it is simple, cost-effective, and light-weight to be implemented on a tag, and also it is practically secure, and highly suitable to the RFID-based supply chain processing scenario. Our scheme provides considerable challenges to thwart malicious readers, disgruntled or compromised employees, and man-in-the-middle attacks.

In section 2 we introduce the one-way reader-to-tag authentication scheme proposed by EPCglobal [4] and describe its security weakness. Section 3 describes our proposed tag-reader mutual authentication scheme. Section 5 provides the security and implementation analysis of our scheme. Section 5 concludes this paper.

2 Related Work

Our proposed scheme is an improvement over the weak One-Way Reader-to-Tag Authentication Scheme proposed by EPCglobal [4]. Therefore in the following subsections we describe this scheme and also its security weaknesses. Table 1 provides the list of notations we used in this paper.

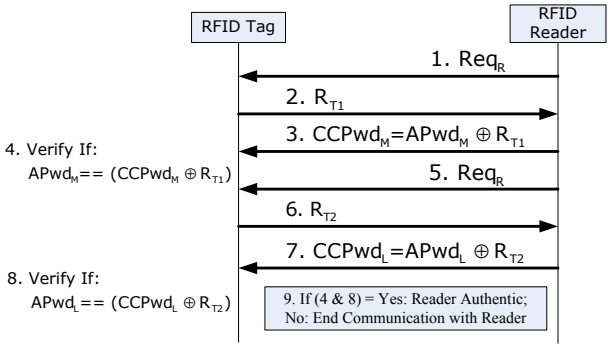

Fig. 1. Security Weakness of EPCglobal Class 1 Gen 2 UHF RFID Protocol

Table 1. Notations.

Notation	**Description**
Req_R	Command Requesting 16bit Random No.
R_{Tx}	16bit Random No. Generated by Tag
R_{Mx}	16bit Random No. Generated by Manufacturer
APwd	Tag's Access Password
KPwd	Tag's Kill Password
$APwd_M$	16 MSBs of APwd
$APwd_L$	16 LSBs of APwd
$CCPwd_M$	Cover-Coded $APwd_M$
$CCPwd_L$	Cover-Coded $APwd_L$
PAD_x	Generated Pads for Cover-Coding
$\|\|$	Concatenates its right operand to the end of its left operand
$\oplus$	Bit-wise XOR Operation

2.1 Security Assessment of EPCglobal Class 1 Gen 2 UHF RFID Protocol [4]

As per EPCglobal Class 1 Gen 2 UHF RFID Protocol standard, a tag's chip has four memory banks: Reserved, EPC, TID, and User. Reserved memory bank is used to store 32-bit *Access Password* (APwd) and 32-bit *Kill Password* (KPwd), and EPC memory bank for EPC number. The reserved memory bank is permanently locked by the manufacturer; therefore APwd and KPwd can neither be read nor modified by any reader. The tag has the capability to verify these two passwords. A reader that presents the right APwd, is allowed to carry out mandatory commands such as Read, Write, and Lock on the tag. If a reader sends the right KPwd, the tag enters the *Killed State*, where it is permanently disabled. The standard does not provide details on how to securely communicate the APwd and KPwd to the readers. Tags can generate 16-bit random or pseudo-random numbers R_{Tx}. While powered, tags can temporarily store at

least two R_{Tx}. Readers and tags implement an *Access* command; which causes the tag 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 command Req_R requesting a random number. Rest of the scheme is fairly easy to understand by studying the multi-step procedure shown in Fig. 1. R_{Tx} is used has XOR pad to obscure APwd, this is known as Cover-Coding APwd (CCPwd). Each XOR operation shall be performed first on APwd's 16-Most Significant Bits (MSB) $APwd_M$, followed by 16-Least Significant Bits (LSB) $APwd_L$.

2.2 Security Weaknesses

- Man-in-the-Middle Attack and *Access Password* Exposed: This scheme is not at all secure, as the tag sends both the R_{Tx} in open and un-encrypted form. Therefore any eavesdropping malicious reader, a disgruntled or compromised employee can easily capture these R_{Tx}, and by carrying out $R_{Tx} \oplus CCPwd$ gives away the APwd. An exposed APwd also allows malicious reader to illegally access, corrupt and modify tag's data.
- Tag Cloning: An exposed APwd would easily assist an adversary to create cloned fake tags with the same APwd.

3 Proposed Tag-Reader Mutual Authentication Scheme

3.1 Supply Chain Processing Scenario and Assumptions

Let us assume that a distributor receives a pallet of products from a manufacturer. The distributor must authenticate the tag attached to the pallet. But the reader at the distributor's end does not know the tag's APwd. Therefore the reader contacts the manufacturer in order to get the APwd. But, giving away the APwd to the distributor would compromise the security of the tag for the rest of its product life cycle and supply chain processing. Therefore in our scheme the *Manufacturer*, Distributor's *Reader*, and the *Tag* follow a multi-step tag-reader mutual authentication procedure (Shown in Fig. 2).

As per the EPCglobal Architecture Framework Specification [3], RFID readers are supported, monitored, and managed by many back-end computer terminals and programs such as RFID Middleware, EPCIS Accessing Application, EPCIS Query Interface, EPCIS Repository, and ONS. For reasons of clarity, we will consider the readers at the distributor's end and their back-end computer terminals and programs as one single entity called: "RFID Reader". We assume that the communication channel between the resource rich entities like *RFID Reader*, and *Manufacturer*, to be highly secure (SSL-TLS, EAP-TLS, and X.509 Authentication Framework). The trusted "Subscriber Authentication [3]" core service identifies the roles (distributor, wholesaler, and retailer) of various stakeholders and distributes appropriate X.509 type certificates [5] to them. These certificates authenticate, authorize, and secure the communication channel

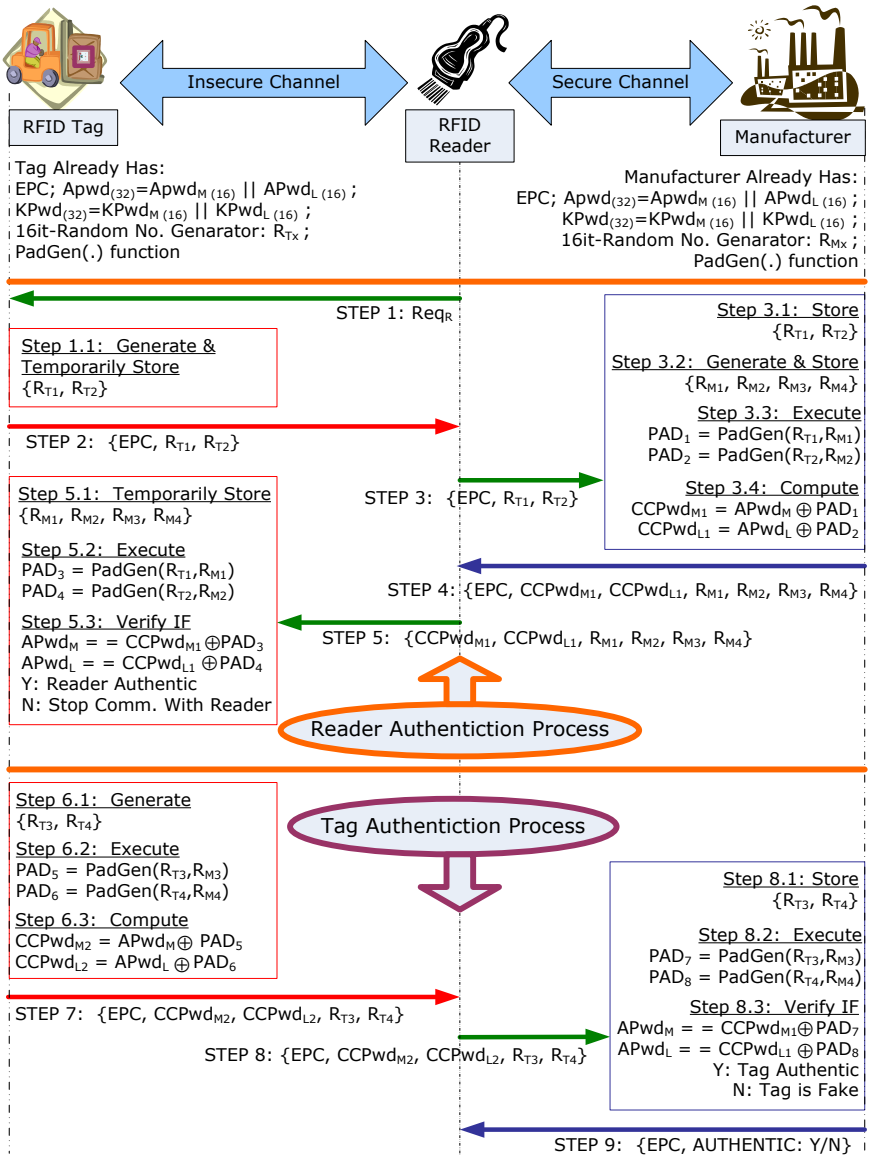

Fig. 2. Proposed Tag-Reader Mutual Authentication Scheme

among them. We assume that the *RFID Reader*, and *Manufacturer* share their digital certificates and be able to execute Signature - $Sig\{.\}$ and Encryption - $Encr\{.\}$ functions.

3.2 Description of the Proposed Scheme

Our proposed scheme can be easily understood by looking at Fig. 2. Steps 1-5 details Reader Authentication Process. Steps 6-9 describe Tag Authentication Process. Please note that Steps 1-9 are carried out in one interrogation session between the tag and the reader. After Step 9, if the verification of the tag is successful, the *Manufacturer* also updates it's EPC-IS Repository with the information that a pallet to which this authentic tag is attached to has reached the distributor, and also other information associated with this event.

One of the main components of our proposed scheme is PadGen(.): *Pad Generation Function*. Detailed description of the PadGen(.) function is described in the next sub-section. In short, this function takes two 16-bit random numbers each, from the Tag (R_{Tx}) and the Manufacturer (R_{Mx}), and utilizes the *Access* (APwd) and *Kill* (KPwd) *Passwords*, to generate two 16-bit Pads (PAD_x). Since ONLY the tag and the manufacturer know (APwd) and (KPwd), just by sharing the random numbers among themselves (via RFID reader), both the tag and the manufacturer can generate the same pads. Later these two pads are in-turn used to cover-code (XOR) the two 16-bit (APwd) chunks ($APwd_M$, $APwd_L$). This approach prevents the major drawback of the one-way reader-to-tag authentication scheme proposed by EPCglobal, where the random numbers sent in open, un-encrypted form, are used as pads to cover-code the APwd chunks. But in our proposed scheme the generated pads are known only to the tag and the manufacturer, and using them to cover-code the APwd chunks, provides fair amount of obscurity and security to the real APwd. Therefore we can fend off threats like exposed tag's APwd, malicious snooping readers, disgruntled employee, man-in-the-middle attacks, and cloned tags.

The manufacturer and the reader mutually authenticate and authorize each other via their digital certificates and signatures. The manufacturer sends the cover-coded ($APwd$) chunks $CCPwd_{M1}$, $CCPwd_{L1}$ to only authorized and authenticated reader. When the reader presents $CCPwd_{M1}$, $CCPwd_{L1}$ to the tag, the tag verifies them and if tallied the tag authenticates the reader to be genuine. Manufacturer Authenticates Reader, Tag Authenticates Manufacturer, therefore Tag Authenticates Reader. Similarly the tag sends $CCPwd_{M2}$, $CCPwd_{L2}$ to the reader, the reader passes them on to the manufacturer, where they are verified and if tallied, the manufacturer informs the reader that it is handling a genuine tag. Reader Authenticates Manufacturer, Manufacturer Authenticates Tag, therefore Reader Authenticates Tag.

3.3 Pad Generation Function - PadGen(.):

Formula:

- $CCPwd_M = APwd_M \oplus PAD$
- $PAD = PadGen(R_{Tx}, R_{Mx})$
  $= KPwd - PadGen(APwd - PadGen(R_{Tx}, R_{Mx}), R_{Tx})$

Let us represent the 32-bit APwd as:

- Hexadecimal (Base 16) Notation: $A \in \{0, 1, 3, \cdots, 9, A, B, C, \cdots, F\}$
  APwd $= A_0A_4A_8A_{12}A_{16}A_{20}A_{24}A_{28}$
- Binary (Base 2) Notation: $a \in \{0, 1\}$
  APwd $= a_0a_1a_2a_3a_4a_5a_6 \cdots\cdots a_{28}a_{29}a_{30}a_{31}$
  APwd $= APwd_M \| APwd_L$
  $APwd_M = a_0a_1a_2 \cdots\cdots a_{13}a_{14}a_{15}$
  $APwd_L = a_{16}a_{17}a_{18} \cdots\cdots a_{29}a_{30}a_{31}$

Let us represent the 32-bit KPwd as:

- Hexadecimal (Base 16) Notation: $K \in \{0, 1, 3, \cdots, 9, A, B, C, \cdots, F\}$
  KPwd $= K_0K_4K_8K_{12}K_{16}K_{20}K_{24}K_{28}$
- Binary (Base 2) Notation: $k \in \{0, 1\}$
  KPwd $= k_0k_1k_2k_3k_4k_5k_6 \cdots\cdots k_{28}k_{29}k_{30}k_{31}$
  KPwd $= KPwd_M \| KPwd_L$
  $KPwd_M = k_0k_1k_2 \cdots\cdots k_{13}k_{14}k_{15}$
  $KPwd_L = k_{16}k_{17}k_{18} \cdots\cdots k_{29}k_{30}k_{31}$

Let us represent the 16-bit random number R_{Tx} generated by Tag as:

- Hexadecimal (Base 16) Notation: $ht \in \{0, 1, 3, \cdots, 9, A, B, C, \cdots, F\}$
  $R_{Tx} = ht_1ht_2ht_3ht_4$
- Decimal (Base 10) Notation: $dt \in \{0, 1, 3, \cdots, 9, 10, 11, \cdots, 15\}$
  $ht_i = dt_i$
  $R_{Tx} = dt_1dt_2dt_3dt_4$

Let us represent the 16-bit random number R_{Mx} generated by Tag as:

- Hexadecimal (Base 16) Notation: $hm \in \{0, 1, 3, \cdots, 9, A, B, C, \cdots, F\}$
  $R_{Mx} = hm_1hm_2hm_3hm_4$
- Decimal (Base 10) Notation: $dm \in \{0, 1, 3, \cdots, 9, 10, 11, \cdots, 15\}$
  $hm_i = dm_i$
  $R_{Mx} = dm_1dm_2dm_3dm_4$

Let us compute: $APwd - PadGen(R_{Tx}, R_{Mx})$

- $APwd - PadGen(R_{Tx}, R_{Mx})$

  $= a_{dt_1}a_{dt_2}a_{dt_3}a_{dt_4} \| a_{dt_{1+16}}a_{dt_{2+16}}a_{dt_{3+16}}a_{dt_{4+16}} \|$
  $a_{dm_1}a_{dm_2}a_{dm_3}a_{dm_4} \| a_{dm_{1+16}}a_{dm_{2+16}}a_{dm_{3+16}}a_{dm_{4+16}}$ [Base 2]

  $= hv_1hv_2hv_3hv_4$ [Base 16, where $hv \in \{0, 1, 3, \cdots, 9, A, B, C, \cdots, F\}$]

  $= dv_1dv_2dv_3dv_4$ [Base 10, where $dv \in \{0, 1, 3, \cdots, 9, 10, 11, \cdots, 15\}$]

Let us compute: $KPwd - PadGen(APwd - PadGen(R_{Tx}, R_{Mx}), R_{Tx})$
$= KPwd - PadGen(hv_1hv_2hv_3hv_4, R_{Tx})$

- $KPwd - PadGen(hv_1hv_2hv_3hv_4, R_{Tx})$

  $= k_{dv_1}k_{dv_2}k_{dv_3}k_{dv_4} \| k_{dv_{1+16}}k_{dv_{2+16}}k_{dv_{3+16}}k_{dv_{4+16}} \|$
  $k_{dt_1}k_{dt_2}k_{dt_3}k_{dt_4} \| t_{dt_{1+16}}k_{dt_{2+16}}k_{dt_{3+16}}k_{dt_{4+16}}$ [Base 2]

  $= hp_1hp_2hp_3hp_4$ [Base 16, where $hp \in \{0, 1, 3, \cdots, 9, A, B, C, \cdots, F\}$]

$\therefore \quad PAD = hp_1hp_2hp_3hp_4$ [Base 16]

4 Analysis of Our Proposed Scheme

4.1 Security Analysis

Our proposed scheme is not fully secure, it suffers from the fact that the APwd and KPwd are only 32-bits each (as per the EPCglobal standard). A simple brute-force attack or other active-attacks on the tag, can crack these two passwords. This is a trade off between keeping our scheme simple, low-cost, and adhering to EPCglobal standards, instead of proposing an expensive and a completely secure scheme. But for a RFID-based supply chain processing scenario, our proposed scheme proves to be light-weight and practically secure, this aspect is highlighted in the following sections. Our scheme provides tag-reader mutual authentication scheme, and prevents the leakage of tag's APwd by the stakeholder's reader or by a disgruntled/compromised employee.

Practically Secure:

An active attacker may continuously eavesdrop on the communication channel between a particular tag and a reader, in order to extract that tag's APwd and KPwd. Since both the passwords are only 32-bits, the attacker can easily mount a ciphertext-only attack. Such active attacks can be prevented by processing the tagged items in an enclosure (warehouse) that is sealed off from external noise and radio signals from malicious readers. In an extremely fast paced, RFID supply chain processing environment, it is not feasible to continuously eavesdrop on one particular tag-reader communication channel for a time long enough to mount ciphertext-only attack. Several bulks of items pass through the readers with in a very short interval of time.

Tag-Reader Mutual Authentication:

Reader Impersonation Attack: The first phase of our proposed scheme is for the reader to authenticate itself to the tag. But a malicious reader does not posses both the APwd and KPwd, in order to generate corresponding CCPwd. The tag can easily detect a false CCPwd and immediately stop communication with the malicious reader. A malicious reader cannot even access the manufacturer (EPC-IS) due to lack of authenticating and authorizing credentials. Therefore a Genuine Reader Impersonation Attack cannot be successful.

Cloned Fake Tags and Tag Impersonation Attack: The second phase of our proposed scheme is for the tag to authenticate itself to the manufacturer. But a malicious tag or a cloned fake tag, do not posses both the APwd and KPwd, in order to generate corresponding CCPwd. The manufacturer can easily detect a false CCPwd and notify the reader that the tag in question is not authentic, it could be either a fake tag or a malicious tag.

On the other hand, a fake tag or a device emulating the functionalities of a malicious tag, may use the same random numbers or weak random numbers (*e.g.,* 0000_h, 1111_h, $FFFF_h$, etc.) repeatedly in order to cryptanalyze the CCPwd obtained from the manufacturer (during the reader authentication phase of our scheme). Therefore for additional security, the reader (at the distributor's end)

or the manufacturer must detect and terminate the communication, if one particular tag is using the same or weak random numbers for over a certain number of consecutive sessions. Since the manufacturer is a resource rich entity, it can keep track of the random numbers and also enforce the generation of good quality random numbers from the tag. The reader or the manufacturer can easily detect an anomaly, if one particular tag is being interrogated or making its presence felt more than a certain pre-defined number of times. This means that this tag is stationary, and is not moving through the supply chain processing. Chances are that, it can be a device emulating the functionalities of a malicious tag. With the above two security measures a Genuine Tag Impersonation Attack cannot be successful.

Tag's Access Password Never Exposed:

Unlike the EPCglobal's authentication scheme, our scheme does not use the random numbers sent in an un-encrypted form as pads to cover-code the tag's APwd. Instead these random numbers are used in association with the tag's APwd and KPwd to generate the pads. These generated pads are known only to the tag and the manufacturer. Using these pads to cover-code the APwd provides fair amount of obscurity and security to the tag's real APwd.

Secure against Insider Attacks:

In order to prevent leakage of APwd by disgruntled/compromised employees or readers, our proposed scheme does not deliver the tag's APwd to any of the stakeholder's reader. The reader (*e.g.,* at distributor's end) relays only the cover-coded APwd from both the Manufacturer, and the tag. Only the tag and the manufacturer can compute the right pads to verify the CCPwd. We can also adopt a "RFID system level check", where the system gives out an alert to the manufacturer, whenever a particular compromised reader at a stakeholder's location is continuously trying to interrogate only one particular tag with an intention to crack its APwd.

Secure against Replay Attacks:

To compute the pads, we use two random numbers each, generated by both the tag and the manufacturer. Therefore replaying a particular session would not serve any purpose for the adversary, as at least either the tag or the manufacturer would be genuine to generate unique random numbers for every session. As unique random numbers are used during different sessions, the computed pads are always unique.

Password Scalability:

As mentioned before, a 32-bit password is not secure against active attacks like brute-force attack or ciphertext-only attack. We did not want to make major changes to the ratified standard, so we adhered to the 32-bit passwords and enhanced its security with very minor tweaks. Our proposed scheme can still be applicable, and more strengthened, in the case, where the length of the APwd and KPwd is extended for active-tags or tags for very expensive items.

4.2 Implementation Analysis

In section 3.1, we assumed that, in order to secure their communication channel the *RFID Reader*, and *Manufacturer* share their digital certificates and be able to execute Signature - $Sig\{.\}$ and Encryption - $Encr\{.\}$ functions. These PKI-based certificate, encryption and signature schemes are expensive w.r.t computational and performance factors. One may also feel that our proposed scheme may cause overhead to the RIFD-based supply chain management system, as the stakeholder's reader needs to securely communicate with the manufacturer in order authenticate every tag.

To reduce this overhead, the manufacturer can setup a secure server at every stakeholder's supply chain processing facility. Only, the manufacturer can remotely access, monitor, and manage this server and also update the server with tags' Access & Kill passwords, and other required data. The stakeholder's RFID reader can now securely query this server in order to authenticate any tag in it's possession. We can also assume that the manufacturer's EPC-IS is a highly resource rich entity, which is designed to take heavy computational and storage load. EPC-IS is actually a network of high performance computer terminals and huge databases, whose main role is to assist a very large number of supply chain partners and consumers. We therefore assume that the manufacturers must have installed load balancing, firewall, bandwidth management, and backup mechanisms to support EPC-IS. If the above assumption is not feasible for some reasons, during the first PKI-based authentication and encryption, reader and manufacturer can share a symmetric key. After which, we can secure the communication channel with only Keyed-Message Authentication Code (MAC), which reduces a great deal of burden.

Light-Weight Tag-Reader Mutual Authentication:

Our scheme does not use any special cryptographic functions. As per the EPCglobal Class 1 Gen 2 UHF RFID Protocol standard [4], the tag has the capability to compute XOR operations, generate random numbers, temporarily store random numbers and fetch the APwd and KPwd embedded within its Reserved Memory bank. Our scheme utilizes only these features.

Our scheme just needs an additional five 16-bit temporary storage memory slots within the tag, for four random numbers from the manufacturer and one for PadGen(.) function. Since Class-1 Gen-2 tags can have a 512-bit memory capacity or more (depending on the manufacturer), these additional five 16-bit temporary storage memory slots, can be easily incorporated. The one-way reader to tag authentication scheme proposed by EPCglobal requires two 16-bit temporary storage memory slots. Pad generation function utilizes the tag's (already existing) memory fetch capability, which collects the individual bits of the APwd and KPwd from the memory locations identified by the random numbers and concatenates these bits to form PADs. Therefore our proposed scheme is light weight and requires minor changes to the EPCglobal Class 1 Gen 2 UHF RFID Protocol standard.

5 Conclusion

In this paper we identified, that threats from cloned fake RFID tags, malicious snooping RFID readers, and unauthorized tag's data manipulation can only be prevented by incorporating a tag-reader mutual authentication scheme. We also analyzed the security weakness of the one-way reader-to-tag authentication scheme proposed by EPCglobal Class 1 Gen 2 UHF RFID Protocol. We then proposed a simple, cost-effective, light-weight, and practically secure tag-reader mutual authentication scheme that adheres to EPCglobal standards. Our scheme utilizes only the XOR operation, and tag's access password and kill password for achieving tag-reader mutual authentication. Therefore, in our scheme, the tag's access password is never exposed even to the stockholder's reader (protection from insider attacks), yet we accomplish tag-reader mutual authentication. The manufacturer of the product also plays a vital role in the mutual authentication procedure and as a result, the manufacturer can immediately know that a particular genuine tag attached to a product (container, pallet, carton, case, and item) has reached the intended stakeholder. In our future work we will formally prove the security of our proposed scheme and analyze its performance on a RFID-based supply chain test-bed.

References

1. VeriSign, "The EPCglobal Network: Enhancing the Supply Chain", White Paper 2005, `http://www.verisign.com/stellent/groups/public/documents/white_paper/002109.pdf`
2. EPCglobal Web site, 2005, `http://www.EPCglobalinc.org`
3. EPCglobal Specification, "The EPCglobal Architecture Framework", `http://www.epcglobalinc.org/standards/`
4. EPCglobal Ratified Standard, "EPC^{TM} Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz - 960MHz Version 1.0.9", `http://www.epcglobalinc.org/standards/`
5. EPCglobal Ratified Standard, "EPCglobal Certificate Profile Standard", `http://www.epcglobalinc.org/standards/`
6. Ari Juels, "RFID Security and Privacy: A Research Survey", RSA Laboratories, 2005.
7. Gildas Avoine and Philippe Oechslin, "A Scalable and Provably Secure Hash-Based RFID Protocol", Proceedings of Workshop on Per vasive Computing and Communications Security PerSec'05, March 2005.
8. P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Rib agorda, "M^2AP: A Minimalist Mutual-Authentication Protocol for Low-cost RFID Tags", Proceedings of Ubiquitous Intelligence and Computing UIC'06, LNCS 4159, pp. 912-923, 2006.
9. Li, Tieyan and Wang, Guilin, "Security Analysis of Two Ultra-Lightweight RFID Authentication Protocols", Proceedings of IFIP SEC 2007, May 2007.