Adaptive Online/Offline RFID Scheme for Supply Chain Management Systems

Zeeshan Bilal
Information Security Group
Royal Holloway University of London
Egham, Surrey, TW20 0EX
Zeeshan.Bilal.2010@live.rhul.ac.uk

Keith Martin
Information Security Group
Royal Holloway University of London
Egham, Surrey, TW20 0EX
Keith.Martin@rhul.ac.uk

***Abstract*—This paper is concerned with RFID tagged objects in a supply chain management system. Such objects are read by multiple readers both in known locations (secure zone with online readers) as well as unknown locations (insecure zone with offline readers). In the secure zone, the primary requirement is to read a large number of tags with high speed. In the insecure zone, the primary requirement is to preserve the privacy of a tagged object. We present an EPCglobal Class-1 Gen-2 Version 1.2.0 standard compliant scheme which allows RFID tags to be authenticated by readers throughout the supply chain lifecycle while meeting the requirements of both the secure and insecure zones.**

***Keywords*-RFID; Supply Chain Management; Online Readers; Offline Readers; Privacy**

I. Introduction

The EPCglobal Class-1 Gen-2 Version 1.2.0 (EPCC1G2) standard [1] specifies low-cost UHF tags which operate in the frequency range of 860-960 MHz and have a read range of 2-10 meters. These tags are typically deployed in supply chain management systems for automated inventory checks. The UHF air interface protocol (explained in Section III-C) defines the standard of communication between a reader and a tag. The reader first selects a group of tags to be read in its vicinity. It then initiates an inventory round to read each tag's content until the whole group is read. Finally it enters into an access phase for writing into a tag's memory if required using a built-in *Access* password.

However, there are privacy issues associated with this class of tags [2]. Since the standard does not elaborate on any specific authentication mechanism, a tag will respond to every query sent by a compatible reader. This causes privacy concerns as follows:

1) **Content Privacy.** An illegitimate reader can learn sensitive information associated with a tag's identifier such as type, price, expiry etc. This can be used to profile the tag holder such as shopping habits, medical history and other private information.
2) **Location Privacy.** An attacker can track a tag carrier since the tag's electronic product code (EPC) is a unique and static identifier.

RFID systems using EPC tags cannot implement computationally intensive privacy-preserving protocols due to their limited resources. EPC tags have limited memory and computation capabilities. These are passive tags and draw power from a reader in order to compute and communicate. In addition, the amount of data transmitted between a tag and a reader should not be excessive, bounded by the available bandwidth.

To better understand the RFID system deployment, we reproduce an example of the supply chain management as given in [3] for illustration. Figure 1 depicts the journey of a pack of razor blades from its manufacturer to a consumer. We start with the manufacturer where one pallet consists of 90 cases with each case containing 72 packs. Therefore considering the pallet, cases and packs are all tagged, a total of 6571 tags reach to a distribution center in one large group. This large pallet is then de-palletized and assembled back into smaller pallets depending on the orders given by retail stores. Considering a smaller pallet can hold up to 10 cases, each pallet will now carry 730 tags stored in the backroom of a retail store. Normally up to two cases are displayed on the store shelf and a consumer may pick few packs to purchase. Following are typical hierarchy of some of the objects:

- *Razor blades:* 6571 → 730 → 144 → 5
- *DVDs:* 5040 → 2520 → 400 → 24
- *Pharmaceuticals:* 7200 → 1920 → 150 → 6

These hierarchies may differ for various objects and retailers. The important point to note here is that the number of tags (tagged items) reduces in size from manufacturer to end-user. The larger group of tags is read by readers in a physically secure environment, whereas as the smaller number of tags, reaching to store shelf and consumers, is exposed to adversaries. Considering a typical supply chain process, we divide the lifecycle of a tag into the following two zones (see example given in [3] for illustration):

1) **Secure Zone with Online Readers.** This zone is assumed to be secure from all adversaries. A large number of tags are scanned by a limited number of known readers in this zone. Since the position of all the readers is known, these readers either share the database held with back-end server which stores shared secrets for each tag, or secrets can be securely transferred to those reader's local databases. The main requirement in this zone is fast reading of the large number of passing tags.
2) **Insecure Zone with Offline Readers.** This zone is assumed to be insecure and open to adversaries. A comparatively smaller number of tags are scanned

by unknown readers here. The position of readers is unknown and their local servers do not share secrets with tags. The main requirement here is to preserve privacy, while it is reasonable to compromise on read speed since the number of tags is smaller.

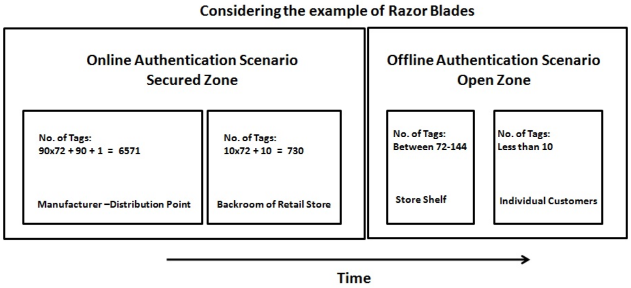

Figure 1. Object's Journey in RFID-enabled Supply Chain Management.

Section II comments on existing approaches to preserving privacy of RFID tags in a supply chain management system. Section III outlines our proposed scheme. Section IV carries out an analysis of our proposal.

II. Existing Work

Various ideas for addressing privacy issues in supply chain management systems have been suggested. Some of these proposals [2], [4], [5] are based on shared secrets (online authentication schemes) and do not address the requirements for tags to be scanned by offline readers. Furthermore, some of these [6], [7] are not EPCC1G2 standard compliant, while some [8], [9] require user intervention in order to preserve the privacy of a tagged object.

A. Password Protected Online Authentication Schemes.

The scheme given in [2] involves disabling RFID tags at checkouts using the existing $Kill$ password. However, secure transfer of $Kill$ passwords to offline readers with unknown locations is not feasible. By disabling tags, after-sales features such as receipt-less returns, automated warranty claims and recycling are not automatically facilitated. The scheme in [4] uses built-in $Kill$ and $Access$ passwords in an EPCC1G2 compliant tag for mutual authentication. While this mechanism avoids killing the tags permanently, a source must know its end destination in order to transfer corresponding passwords. Thus, readers must know all the passwords of potential tags, which could be millions in number, and thus requires a dedicated database. A small retail store cannot afford the luxury of a back-end database and an end-user cannot carry IT equipment in order to transfer all the passwords related to their tags. The proposal in [5] suggests using pseudonyms instead of the original identifiers of tags. However, fixed pseudonyms facilitate tracking, whereas cryptographically changing pseudonyms require readers to possess the same key and stay synchronized. Moreover, a central repository storing all pseudonyms requires access tokens. All of these schemes thus only work with online readers.

B. Additional Privacy Preserving Devices.

Another scheme proposed in [10] uses appropriate prefixes to EPC and an additional blocker tag to preserve the privacy of tags. For example, all the tags attached to sold items are declared to be private (no reader can query the tag) by setting their EPC's prefix bit(s) to some predetermined value. If an unauthorized reader queries these tags, the blocker tag, acting as intermediary, suppresses its queries. As well as requiring an additional blocker tag, this scheme also requires writing/setting the appropriate prefix into a tag's EPC memory (for example at point of sale). This scheme is based on querying a tag using a binary-tree search algorithm and is not EPCC1G2 compliant.

The proposals given in [11]–[13] use a proxy device to suppress the stealth scanning of a tag's content. The proxy device acts as an intermediary between reader and tag. This smart device makes intelligent decisions in determining the legitimacy of a reader. However in these proxy devices, acquire and release control of tags during ownership transfer is difficult. It is also difficult to entirely suppress reader's commands and tag's replies.

C. Distance Bounding Protocols.

There are many proposals for distance bounding protocols [7], [14], [15] which determine the legitimacy of a reader based on its proximity, typically calculated from signal strength and query-to-response time measurements. However, since the read ranges vary considerably depending on the transmitted powers, antenna sensitivities and environment, the adversary may send a stronger signal than prescribed and read over a longer distance with a better signal-to-noise ratio. Therefore, these schemes can fail against such attacks. Moreover these protocols typically require additional circuitry in low-cost tags and are not EPCC1G2 compliant.

D. Relabeling and Partial Destruction.

Similarly some proposals suggest partial destruction of important and secret information of a tag. Relabeling [8] is one such proposal which requires changing the tag's label from secret to some public value in order to preserve the tag's privacy when the tag travels in the insecure zone. Partial destruction using splitting [9] requires two tags (one carrying the private information while the other has public information) on every item. The tag carrying secret information is removed to preserve the privacy when in the insecure zone. Both of these schemes require user interaction.

E. Bit Throttling and Secret Sharing Schemes.

To deter sporadic reading of a tag's secret content, the scheme in [6] reveals the secret content one bit at a time and thus delays the process of promiscuous reading of the tag's content. This makes it harder for a sporadic adversary to disclose or track a particular tag. However the data rate of this scheme is very low and it also requires additional circuitry to perform this task. Determining the sequence of bits for transmission is also a problem as sequential transmission (starting from the least significant bit) can reveal important information through only the first few disclosed bits (for example, the first four bits of the EPC reveal the commercial code and the next four suggest the size).

The scheme suggested in [3] adopts secret sharing where shares are distributed amongst different tags across time and space. When individual tags are sold to different customers, their privacy is preserved as an individual share does not reveal any sensitive information. However, warranty claims become cumbersome in these scenarios because an individual customer carries only one share of the secret and also needs to collect other shares which are distributed amongst other unknown customers. Another potential problem with this scheme is clandestine tracking as secret shares are static and do not change.

The proposal in [16] is based on delayed transmission of the secret value using linear feedback shift registers (LFSR). This proposal is suitable in scenarios where the number of tags is small as it takes time to transmit the complete secret. It therefore does not address the requirement of high speed reading of a large number of tags in the secure zone. It also requires additional functionality other than the standard.

F. Our Scheme.

In this paper, we consider taking an EPCC1G2 compliant approach that fulfills the requirements of both fast read speed when a large number of tags are read by online readers in the secure zone, as well as preserving the privacy of a tag when read by offline readers in the insecure zone. Our unified scheme is based on delaying the disclosure of the secret until a certain time threshold is achieved and adapts between online and offline authentication without user intervention. We focus our comparative analysis as shown in Table I on the schemes presented in [3], [6], [16] since these are the only other schemes which use related techniques.

III. Proposed Scheme

We now explain our proposed scheme which provides privacy to EPCC1G2 compliant RFID systems deployed in a supply chain management system.

A. Adversarial Model

We make the following assumptions about the capability of an adversary:

1) An adversary can conduct both passive and active attacks. Our scheme protects against passive attacks (eavesdropping both the forward and backward channels) and active attacks except for physical capture and tampering attacks.
2) An adversary cannot take over an ongoing authentication round because when the tag receives queries from multiple readers, it detects a collision and stops responding (we assume the use of a reader anti-collision algorithm, see [1]).
3) An adversary cannot learn the update values of $RN16$ and $Access$ password (refer to Table II) as only a legitimate reader in possession of the tag can update its memory.

The notation required is shown in Table II.

B. Goals

Considering a supply chain process consisting of the two zones identified in Section I, our scheme is designed to achieve the following goals in the presence of an adversary as defined in Section III-A.

1) **Content Privacy:** Support privacy of a tag's content, wherever this is required.
2) **Location Privacy:** Support privacy of the location of a tag in order to prevent tracing and tracking of the tag, wherever this is required.
3) **EPCC1G2 Compliance:** Fit into low-cost EPCC1G2 compliant tags.
4) **Fast Read Speed:** Support a fast read speed, wherever this is particularly required when the number of tags is large.
5) **User Transparency:** Adapt according to the status of the reader (i.e., online or offline) without user intervention.

C. Overview of Protocol

We use the existing functionality of EPCC1G2 standard tags [1]. The standard defines the ultra high frequency (UHF) air interface protocol shown in Figure 2. We now give an overview of our proposed protocol. Note that we need to make a couple of very minor changes to the standard in order to support an authentication mechanism (see also Section IV-C).

1) **Initialization.** In the original standard [1], each tag generates a random 16-bit number $RN16$ on the fly. We suggest that each tag is initialized using a unique random $RN16$ in its local group. It is important to note that this limits only a group size to 2^{16} tags and does not affect the EPC which is $96-bit$ unique code. This modification can easily be incorporated into the standard. Initially manufacturers can write this into the tag's memory and later the server, in possession of the corresponding $Access$ password, can update the value of $RN16$ by writing into the tag's memory, using a compatible reader. Since a server keeps updated record of groups of tags, it can ensure unique allocation of updated $RN16$.
2) **Initial Identification.** This unique random $RN16$ is used to identify a tag in the reader's back-end server.
3) **Mutual Authentication.** We incorporate a mutual authentication stage inside the inventory round (see [1]). The standard defines two secret values, $Kill$ and $Access$ passwords, that are embedded into every EPCC1G2-compliant tag. The $Kill$ password is used for disabling a tag and the $Access$ password is used for read/write access to the tag. Both passwords are 32-bits long. We use the $Access$ password for both mutual authentication and read/write access, while retaining the $Kill$ functionality where required. We divide the $Access$ password into two

Table I
COMPARATIVE ANALYSIS PROPOSED SCHEME VS EXISTING SCHEMES

Security Features	Marc [6]	Juels [3]	Amariucai [16]	Proposed
Unified Approach	No	No	No	Yes
EPCC1G2 Compliance	No	No	No	Yes
Read Speed	Slow	Fast	Slow	Fast(secure zone)
Content Privacy	Reveals pattern	Preserved	Preserved	Preserved
Location Privacy	Preserved	Not preserved	Preserved	Preserved
Information leakage	Gradual	Gradual	Linear	No Leakage

Table II
NOTATION

Notation	Description
$Query$	A command sent by the reader to a tag/group of tags it wants to read.
$QueryRep$	A command sent by the reader to a tag/group of tags if it receives no response, or multiple of responses from more than one tag.
$SlotCounter$	A counter implemented in the tag which loads a random number and decrements with every $Query$ and $QueryRep$ command.
$RN16$	A 16-bit random number generated by the tag and transmitted to the reader once its $SlotCounter$ reaches zero.
ACK	A 16-bit acknowledgment sent by the reader to the tag.
$PC + EPC + CRC$	A tag's content plus its cyclic redundancy check.
$Access(0 : 15)$	First 16 bits of the unique built-in 32-bit access password in each tag starting from the least significant bit.
$Access(16 : 31)$	Last 16 bits of the unique built-in 32-bit access password in each tag starting from the least significant bit.
$Rand$	A 16-bit random number generated by the tag.
$Rand_n$	The n^{th} $Rand$ generated by the tag.
$Rand_{Th}$	The $Rand$ generated by the tag when a certain time threshold is achieved.

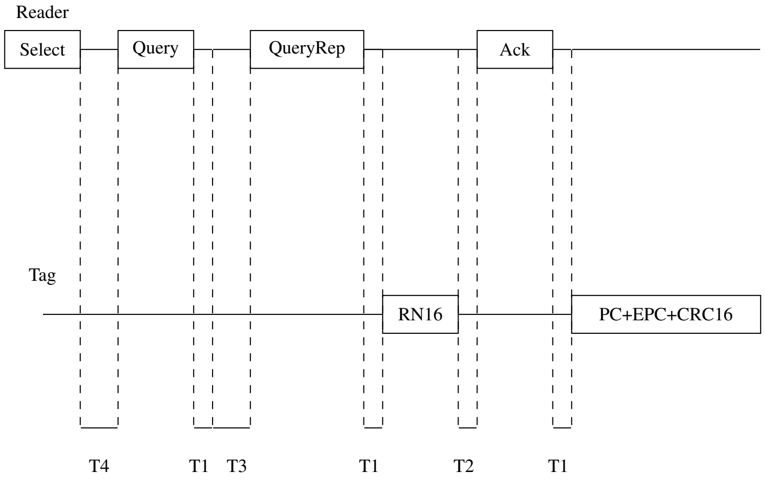

Figure 2. UHF Air Interface Protocol for Class-1 Gen-2 Tags.

parts consisting of the 16 least significant bits (used for reader authentication) and the 16 most significant bits (used for tag authentication).

4) **Standard Protocol.** After successful mutual authentication, tags are read as per the standard [1], as shown in Figure 2.
5) **Update.** We use the access round (see [1]) to enable a legitimate reader to update the values of $RN16$ and the $Access$ password by writing into the tag's memory. Note that this update can only be carried out by a server in possession of the tag's $Access$ password.
6) **Determining Threshold.** Our offline authentication stage is based on a time threshold value (as will be explained in Section III-E). Therefore it is important to determine a suitable threshold value which prevents an adversary from disclosing the contents of the tag or identifying its location. As per the

standard, the reader powers up the tag, sends the select and query commands, receives the response ($RN16$) from the tag, and then transmits an ACK in response. If the ACK is valid, the tag answers back by transmitting its content. The reader then powers down the tag. The whole process ignoring the proposed mutual authentication messages takes approximately 35 milliseconds (see [1]). A legitimate offline reader does not power down the tag until the required time threshold is achieved. Considering power down time is 1 millisecond, one cycle of the standard scanning process without powering down the tag will take approximately 34 milliseconds. Consider a realistic scenario for supply chain management systems where legitimate offline readers are present in retail stores and smart home appliances. These readers can scan the tags for a relatively long time and then change their status to online after obtaining the shared secret when the time threshold is achieved. The precise time threshold value can be set by a manufacturer depending on the application.

The overview of the protocol is shown in Figure 3. Our scheme starts when a reader sends the acknowledgment, which is compared with the value of the $Access$ password stored in the tag. If it matches, our online part of the authentication scheme takes over, otherwise it switches to offline mode. Since reader is only an intermediary device between a server and a tag, each reader is connected to either a back-end server with stored shared secrets with the tag (online readers) or local servers without any information about tags. This connection between reader and server is assumed to be secure, hence we use the term of only reader to encompass reader, server and their communicating channel. Both the online and the offline mechanisms are explained in subsequent sections.

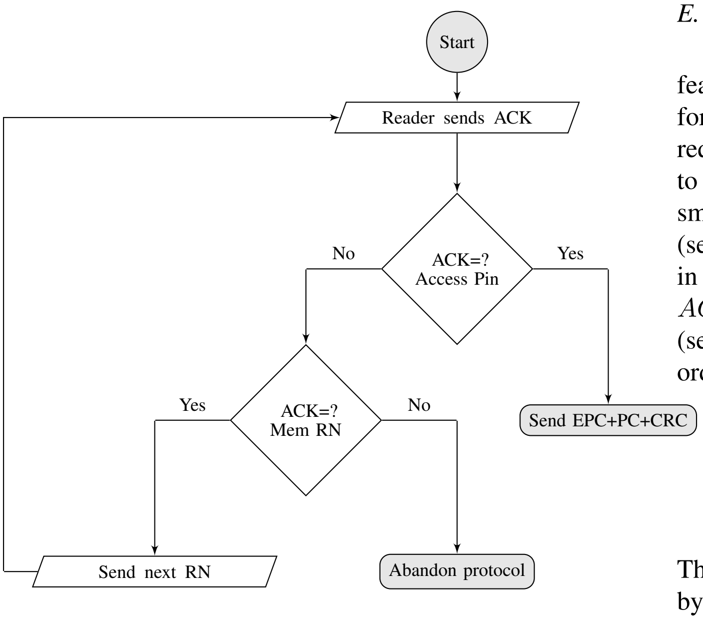

Figure 3. Overview of the Proposed Scheme.

D. Online Authentication Stage

Online authentication is based on shared secrets. Online readers have known locations, and secret passwords ($Kill$ and $Access$) for each tag are securely distributed to every reader in the chain (more precisely all readers share the database storing secret passwords of each tag). The main requirement here is to achieve a fast read rate, since the number of tags is large and the area is considered to be physically secure (see Section I). Since the UHF Air Interface Protocol does not define any authentication mechanism [1], we modify the standard functionality by changing the $RN16$ sent by the tag and the ACK sent by the reader to achieve mutual authentication. Our online authentication scheme is motivated by [4] and defined as follows:

1) **Initialization.** Each tag is initialized with a unique $RN16$.
2) **Initial Identification.** Online readers identify a particular tag using $RN16$ as an index to its database.
3) **Mutual Authentication.** A valid ACK is now the 16 LSBs of the $Access$ password. Once a tag receives a valid ACK, the reader is regarded to be online and legitimate. The tag now sends the 16 MSBs of its $Access$ password, which the reader uses to authenticate the tag.
4) **Standard Protocol.** After successful mutual authentication, the standard as shown in Figure 2 is followed. The reader sends a standard ACK (which is the same $RN16$ sent initially by the tag) and the tag in return sends its information to the reader.
5) **Update.** The legitimate reader updates $RN16$ and $Access$ password values in the tag securely (considering the reader is now in possession of the tag).

The online authentication scheme is summarized in Figure 4, assuming the protocol follows the standard until the slot counter of a particular tag reaches zero.

E. Offline Authentication Stage

Offline readers have unknown locations and it is infeasible to distribute secret passwords ($Kill$ and $Access$) for each tag securely to every such reader. The main requirement here is to preserve privacy with a willingness to compromise on read speed since the number of tags is small and the area is considered to be physically insecure (see Section I). The UHF Air Interface Protocol works as in the standard except that the $RN16$ sent by the tag and ACK sent by the reader changes in the proposed scheme (see Section III-C). The ACK is checked by the tag in order to establish which of the following three states apply:

1) *Valid* if ACK is equal to the 16 least significant bits of the $Access$ password.
2) *Semi-valid* if ACK is equal to the random values generated by tag.
3) *Invalid* otherwise.

The offline part of our authentication scheme is motivated by [16]. This scheme is defined as follows:

1) **Initialization.** Each tag is initialized with a unique fixed $RN16$.

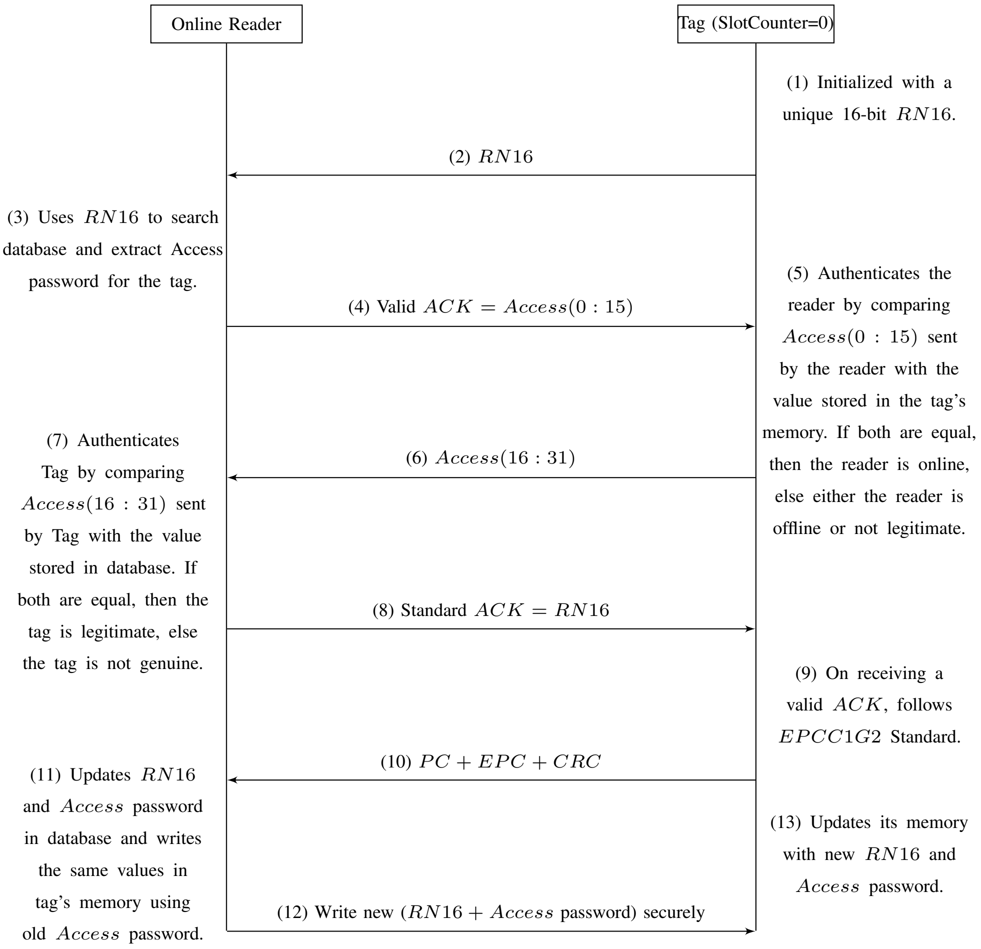
Online Reader
Tag (SlotCounter=0)
(1) Initialized with a unique 16-bit $RN16$.
(2) $RN16$
(3) Uses $RN16$ to search database and extract Access password for the tag.
(4) Valid $ACK = Access(0:15)$
(5) Authenticates the reader by comparing $Access(0:15)$ sent by the reader with the value stored in the tag's memory. If both are equal, then the reader is online, else either the reader is offline or not legitimate.
(6) $Access(16:31)$
(7) Authenticates Tag by comparing $Access(16:31)$ sent by Tag with the value stored in database. If both are equal, then the tag is legitimate, else the tag is not genuine.
(8) Standard $ACK = RN16$
(9) On receiving a valid ACK, follows $EPCC1G2$ Standard.
(10) $PC + EPC + CRC$
(11) Updates $RN16$ and $Access$ password in database and writes the same values in tag's memory using old $Access$ password.
(12) Write new ($RN16 + Access$ password) securely
(13) Updates its memory with new $RN16$ and $Access$ password.

Figure 4. Online Authentication Scheme for Class-1 Gen-2 Tags.

2) **Initial Identification.** Offline readers cannot identify a particular tag using $RN16$, so it cannot send a valid ACK, which is the 16 LSBs of the $Access$ password of the corresponding tag.
3) **Mutual Authentication.** An offline reader sends a semi-valid ACK, which is equal to the $RN16$ (as per the existing standard [1]) sent by the tag. The tag first checks its validity by comparing it with the 16 LSBs of the built-in $Access$ password. In case of failure, it checks its semi-validity by comparing this with the $RN16$ stored in its memory. If the ACK is semi-valid, the tag generates another 16 bit random number $Rand_1$, XORs it with the previous $RN16$, transmits the result Sum_1 to the reader, and stores $Rand_1$ and Sum_1 in its memory (see Figure 5). The reader, on receiving this new Sum_1, stores its value and performs the same operation (i.e. XORs it with the previous value of $RN16$) and sends the result $Rand_1$ to the tag (see Figure 5). The tag continues checking for a valid, semi-valid or invalid ACK and responds accordingly. Once this repeated communication reaches a certain threshold, and the tag determines (by comparing the $Rand_{Th-1}$ sent by the reader with its stored value) that the reader has spent enough time in pairing up, it performs an XOR of the previous value of Sum_{Th-1} stored in its memory with the 16 LSBs of its $Access$ password and sends the result as Sum_{Th} to the reader (see Figure 5). On receiving this 16-bit number, the reader also performs the XOR of this new value Sum_{Th} with the previous one Sum_{Th-1} and extracts the 16 LSBs of the $Access$ password (see Figure 5). Once the reader transmits these 16 bits as an ACK, the tag checks it as valid. On receiving a valid ACK, the tag switches to online mode.
4) **Standard Protocol.** After successful mutual authentication, the EPCC1G2 standard is followed as shown in Figure 2. The reader sends a standard ACK (which is the same $RN16$ sent initially by the tag) and the tag in return sends its content to the reader.
5) **Update.** The legitimate reader updates $RN16$ and $Access$ password values in the tag securely (considering the reader is now in possession of the tag).

The scheme is summarized in Figure 5, assuming the protocol follows the standard until the slot counter of a particular tag reaches zero.

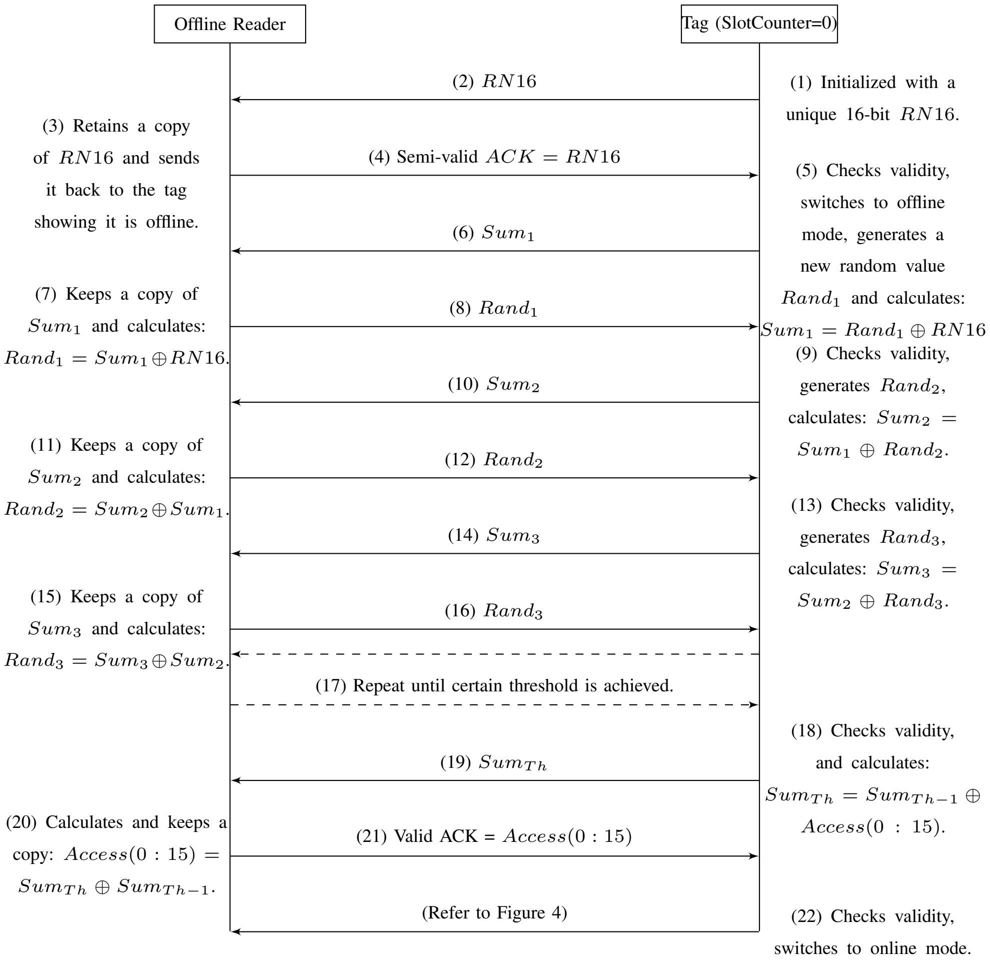
Offline Reader
Tag (SlotCounter=0)
(1) Initialized with a unique 16-bit $RN16$.
(2) $RN16$
(3) Retains a copy of $RN16$ and sends it back to the tag showing it is offline.
(4) Semi-valid $ACK = RN16$
(5) Checks validity, switches to offline mode, generates a new random value $Rand_1$ and calculates: $Sum_1 = Rand_1 \oplus RN16$
(6) Sum_1
(7) Keeps a copy of Sum_1 and calculates: $Rand_1 = Sum_1 \oplus RN16$.
(8) $Rand_1$
(9) Checks validity, generates $Rand_2$, calculates: $Sum_2 = Sum_1 \oplus Rand_2$.
(10) Sum_2
(11) Keeps a copy of Sum_2 and calculates: $Rand_2 = Sum_2 \oplus Sum_1$.
(12) $Rand_2$
(13) Checks validity, generates $Rand_3$, calculates: $Sum_3 = Sum_2 \oplus Rand_3$.
(14) Sum_3
(15) Keeps a copy of Sum_3 and calculates: $Rand_3 = Sum_3 \oplus Sum_2$.
(16) $Rand_3$
(17) Repeat until certain threshold is achieved.
(18) Checks validity, and calculates: $Sum_{Th} = Sum_{Th-1} \oplus Access(0:15)$.
(19) Sum_{Th}
(20) Calculates and keeps a copy: $Access(0:15) = Sum_{Th} \oplus Sum_{Th-1}$.
(21) Valid ACK = $Access(0:15)$
(Refer to Figure 4)
(22) Checks validity, switches to online mode.

Figure 5. Offline Authentication Scheme for Class-1 Gen-2 Tags.

IV. Analysis

In this section, we carry out an analysis of our protocol for the desired goals stated in Section III-B and compare it to existing proposals [3], [6], [16] which are based on a similar mechanism as mentioned in Section II. We summarize this comparison in Table I.

A. Content Privacy

A common criticism of the use of RFIDs is that the tags reveal content promiscuously to any compatible reader. Our scheme protects the content of a tag by only sending them to authorized or trusted readers. In the secure zone with online readers, the tag sends its content only after successful mutual authentication. Considering the area is secured, we rule out the possibility of content disclosure to any adversary. In the insecure zone with offline readers, the tag first sends random information if it does not trust a reader until a certain trust threshold is achieved, then the content of the tag are sent after a successful mutual authentication phase. A recent proposal [16] based on the concept of transmitting a shared secret in parts tends to leak information after every transmission unless the secret is revealed. Our scheme does not reveal any information until the trust threshold is achieved. We analyze the strength of our scheme by considering the following adversarial behaviour:

1) **Online Adversary:** We assume that online readers scan the tag in a secure area (see Section III-A). Therefore, we rule out the possibility of a passive adversary listening to communication between an online reader and a tag. However, an active adversary can act as online (in the insecure area) and the secret $Access$ password can be retrieved by a brute force attack. Simply, a reader can send a random ACK to a tag until the tag sends back its content, which means that the reader has found the correct password. In each guess, the online adversary has to complete the scanning cycle as mentioned above. If the tag does not answer back with its content, the reader powers down the tag and repeats the sequence with a different value of the ACK. Considering the EPCC1G2 specification, each try takes 35 milliseconds and a 16-bit password is thus exhausted in about 38.23 minutes. We consider that an adversary who is not in possession of the tag will not have sufficient time to do this before being detected.
2) **Sporadic Offline Adversary:** A more realistic scenario is of a sporadic adversary who is capable

only of scanning or eavesdropping some of the random information exchanged between a reader and a tag. This random information will not be sufficient to acquire the threshold or disclose the tag's content. Thus the adversary has to keep track of all the communication sessions. However, a sporadic adversary can eavesdrop either the last session (see step 21 onwards in Figure 5), or the second-last session (see steps after the threshold is achieved in Figure 5) by chance. The probability of success will be $1/n$ for a threshold of $n-1$ random sessions since each session is independent. Moreover, the adversary cannot take over an ongoing authentication round (see Section III-A) and has to wait for it to complete. Once an authentication round is complete, the adversary cannot replay the eavesdropped values or act as online since these values are updated in the tag (see Section III-C).

3) **Dedicated Offline Attacker:** A dedicated offline adversary is assumed to act like a legitimate offline reader. This adversary is able to scan the tag until a threshold is achieved. Therefore, the adversary is able to disclose the $Access$ password and content of the tag. After achieving the $Access$ password, the adversary will impersonate as an online reader. It can thus downgrade the legitimate owner to offline by updating the tag to its own values of $RN16$ and $Access$ password. However, if the adversary is not in possession of the tag, this success will be one time only. The adversary will no longer be able to disclose this tag's content since the $Access$ password is updated by the legitimate owner (in its next communication with the tag). The countermeasure for such an adversary is to set the time threshold value to be sufficiently high that this adversary can be detected before the tag reveals its secrets.

B. Location Privacy

Our scheme preserves the location privacy of a tag and hence prevents its tracking. Since $RN16$ and $Access$ password are changed in every authentication round and tag sends different random numbers when queried by an unauthorized reader, its location cannot be tracked. The tracking depends on the properties of random number generator on the tag whose specification are given in the standard [1].

C. EPCC1G2 Compliance

Many of the earlier proposals cannot be implemented in low-cost environments (see Section II), particularly EPCC1G2 standard compliant tags, or require considerable changes to the existing standard. Our scheme can easily be implemented in these tags with very minor changes to the standard and uses existing functionality as defined in the standard [1]. Our scheme does not require any additional functionality because we are using the existing computational capability of the EPCC1G2 standard. However, there are additional communication overheads to achieve mutual authentication and a time threshold. As far as storage is concerned, our scheme requires the tag to store an additional 16-bit value in addition to storing a random number as in the standard. In the online authentication scheme, there is an additional mutual authentication mechanism which is completed in two additional messages and authentication is based on the existing built-in $Access$ password. In the offline authentication scheme, the reader has to acquire a time threshold in order to read the tag's contents. This additional mechanism uses the existing functionalities of an $EPCC1G2$ compliant tag for generating a 16-bit random nonce and conducting an XOR computation.

D. Fast Read Speed

Gen 2 certified readers have 2 read modes: over 1600 tags per second in fast and less than 600 tags per second in slow mode. The read speeds are automatic and depend entirely on the actual read conditions for each tag. In multi-tag environments, where thousands of tags are passing in front of readers, speed is of the utmost importance. Fast read speed requirement exists in the secure zone with online readers. Our proposed scheme reads the tag using the same standard functionality in the secure zone with online readers. Thus this requirement is fulfilled using our proposed scheme.

E. User Transparency

As discussed in Section II, some of the earlier schemes require user intervention to preserve the privacy of the tag. These systems are prone to errors and are labor-intensive. Our proposal adapts between online and offline authentication modes without any user intervention.

V. Conclusion

In this paper, we have proposed a scheme that provides a unified approach to tackle privacy and performance issues in RFID-tagged supply chain management systems. Unlike any existing proposal in the literature, it is easy to implement in the existing EPCC1G2 standard, it provides fast read speed in the secure zone and preserves privacy in the insecure zone, and it adapts between online and offline authentication without user intervention.

References

[1] *EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz-960 MHz, Version 1.2.0*, GS1 EPCGlobal, October 2008, http://www.gs1.org/gsmp/kc/epcglobal/uhfc1g2/uhfc1g2_1_2_0-standard-20080511.pdf.

[2] A. Juels, "RFID Security and Privacy: A Research Survey," *Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 381–394, 2006, iEEE.

[3] A. Juels, R. Pappu, and B. Parno, "Unidirectional Key Distribution Across Time and Space with Applications to RFID Security," in *17th USENIX Security Symposium*. San Jose, California, USA: USENIX, July 2008, pp. 75–90.

[4] A. Juels, "Strengthening EPC Tags against Cloning," in *4th ACM workshop on Wireless security*, ser. WiSe '05. Cologne, Germany: ACM, 2005, pp. 67–76.

[5] ——, "Minimalist Cryptography for Low-Cost RFID Tags," in *4th International Conference on Security in Communication Networks*, ser. Lecture Notes in Computer Science, vol. 3352. Amalfi, Italy: Springer, 2005, pp. 149–164.

[6] M. Langheinrich and R. Marti, "Practical Minimalist Cryptography for RFID Privacy," in *IEEE Systems Journal, Special Issue on RFID Technology*, vol. 1, no. 2. IEEE, December 2007, pp. 115–128.

[7] A. Falahati and H. Jannati, "Application of distance bounding protocols with random challenges over RFID noisy communication systems," in *IET Conference on Wireless Sensor Systems*. London, UK: IET, June 2012, pp. 1–5.

[8] S. E. Sarma, S. A. Weis, and D. W. Engels, "RFID Systems and Security and Privacy Implications," in *4th International Workshop on Cryptographic Hardware and Embedded Systems*, ser. Lecture Notes in Computer Science, vol. 2523. Redwood Shores, CA, USA: Springer, 2003, pp. 454–469.

[9] S. Inoue and H. Yasuura, "RFID Privacy Using User-controllable Uniqueness," in *RFID Privacy Workshop*. MIT, Massachusetts, USA: Citeseer, November 2003, pp. 1–9.

[10] A. Juels, R. L. Rivest, and M. Szydlo, "The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy," in *10th ACM Conference on Computer and Communications Security*, ACM. Washington, DC, USA: ACM Press, October 2003, pp. 103–111.

[11] C. Floerkemeier, R. Schneider, and M. Langheinrich, "Scanning with a Purpose - Supporting the Fair Information Principles in RFID Protocols," in *2nd International Symposium on Ubiquitous Computing Systems*, ser. Lecture Notes in Computer Science, vol. 3598. Tokyo, Japan: Springer, November 2005, pp. 214–231.

[12] M. R. Rieback, B. Crispo, and A. S. Tanenbaum, "RFID Guardian: A Battery-Powered Mobile Device for RFID Privacy Management," in *10th Australasian Conference on Information Security and Privacy*, ser. Lecture Notes in Computer Science, vol. 3574. Brisbane, Australia: Springer, July 2005, pp. 184–194.

[13] A. Juels, P. F. Syverson, and D. V. Bailey, "High-Power Proxies for Enhancing RFID Privacy and Utility," in *Privacy Enhancing Technologies, 5th International Workshop*, ser. Lecture Notes in Computer Science, vol. 3856. Cavtat, Croatia: Springer, 2006, pp. 210–226.

[14] J. S. Kim, K. Cho, D. H. Yum, S. J. Hong, and P. J. Lee, "Lightweight Distance Bounding Protocol against Relay Attacks," *IEICE Transactions on Information and Systems*, vol. 95, no. 4, pp. 1155–1158, April 2012, the Institute of Electronics, Information and Communication Engineers.

[15] S. Lee, J. S. Kim, S. J. Hong, and J. Kim, "Distance Bounding with Delayed Responses," *Journal on Communications Letters*, vol. 16, no. 9, pp. 1478–1481, 2012, iEEE.

[16] G. T. Amariucai, C. Bergman, and Y. Guan, "An Automatic, Time-Based, Secure Pairing Protocol for Passive RFID," in *7th International Workshop on RFID Security and Privacy*, ser. Lecture Notes in Computer Science, vol. 7055. Amherst, USA: Springer, 2012, pp. 108–126.