Protecting the Privacy of RFID tags

by

Nimish Vartak

Thesis submitted to the Faculty of the Graduate School
of the University of Maryland in partial fulfillment
of the requirements for the degree of
Master of Science
2006

Dedicated to my family, friends and my late aunt.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my graduate advisor Dr. Anupam Joshi for his constant support and guidance. I am also grateful to Dr. Tim Finin for his pointers and guidance and my committee members Dr. Yelena Yesha and Dr. Zary Segall.

I would also like to thank Anand Patwardhan and Pranam Kolari who were very enthusiastic about my work and has been a constant help in working on the RFID Privacy aspects. I would also like to thank my labmates and friends at UMBC, who have been helpful in my work.

ABSTRACT

Title of Thesis:

Protecting the Privacy of RFID tags

Author: Nimish Vartak, Master of Science, 2006

Thesis directed by: Dr. Anupam Joshi, Professor
Department of Computer Science and
Electrical Engineering

Radio Frequency Identification (RFID) is an emerging wireless technology with many potential applications, including supply chain management, personnel tracking and point of sale checkout. Its wide spread adoption raises concerns about known security and privacy vulnerabilities, including the ability of rogue RFID readers to access the unique identifier and data of RFID tags. To prevent the eavesdropping of tag through communication channel, methods like one-way hashing, cryptography and one-time pads have been used; however they do not prevent the clandestine tracking of tags using their unique identifier. We describe a novel scheme to protect the identity of tags, and prevent them from being clandestinely tracked and inventoried.

Our approach uses an RFID reader, an authenticating agent, and a local entity that can dynamically reprogram RFID tags to protect their identity. We ensure visibility of goods to authorized RFID readers at any point in the transit of RFID tagged goods from one location to another, while denying information to unauthorized readers. The approach protects the identity of the RFID tags without significant changes to the existing infrastructure. We

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Although RFID can best be described as a technology in its nascent stage, due to strong capabilities of non-line-of-sight operation and unique identification for an item, it is already becoming pervasive and has been extensively deployed for the purposes of tracking goods, and also livestock. From the supply chain management perspective, RFID ensures traceability of goods from cradle to grave (also referred to as the "Holy Grail" using RFID [1]) of goods from manufacturing through sale and even beyond it. EPC Global Inc. [2] has developed the EPC Global Network [3] which has services to uniquely identify any RFID tagged item, and get the tag information. The tag's unique identifier acts as a pointer to the item's details in a secure database. The database security and access controls do not however prevent the item from being tracked by an adversary based solely on its unique ID. The RFID tag's unique identifier typically is a key in a database table which holds more details of the item. Instead of trying to compromise the database access controls, an adversary could find alternate means to determine the details of the tags. RFID readers are easily available and, in fact, certain cell phones are equipped with RFID readers [4]. Hence, it is possible for an adversary to read the RFID tag and correlate its time and place to learn more about the tag. Tracking a tag bearer is possible today with many leading clothing manufacturers [5] and shipping companies [6] using RFID tags. This problem of clandestine tracking [7] is not overcome even if another (unrelated) identifier is stored onto

the tag, so long as the identifier remains static. The privacy problem gets more complicated if the tag itself holds details of the item like the price. An adversary may eavesdrop on a genuine reader-tag communication including the singulation[1] [8] protocol. A variety of schemes have been proposed to protect this communication. These include encryption mechanisms [9], hashing [10], and use of one-time pads [11]. The problem is more aggravated for basic passive (identity EPC class 1 [11]) RFID tags, as a basic interrogation of a RFID tag by an RFID reader may be sufficient to reveal this unique identifier. Although the most recent specification of the EPC Class 1, Generation 2 [11] specifies access-control mechanisms for tags; the access control is meant for the ability to write or kill a tag but does not prevent a tag from being read.

To solve this problem of unauthorized read access, ideally each tag should be capable of computation and be able to choose whether or not to identify itself, based on credentials from the interrogating RFID reader. Further, depending on the access level granted to the reader, a tag could possibly provide different granularity of information. This may include selectively revealing bits of the identification, with the remaining bits masked off. However, this requires considerable computation capability on part of the tag. The most common and inexpensive passive tags have very limited computational capabilities. Hence, this approach would require the more expensive active tags which are capable of computation. Since such a solution would not be economically viable, our focus is on using existing technology and passive tags. We also describe a security enhanced variant of our scheme using active tags. We introduce a local entity to enforce the security for tags in the vicinity, so as to ensure that information is readily available to authorized sources, while preventing its abuse by unauthorized entities. We explain our scheme in context of securing the identity of valuable items in transit. While the EPC Global Network caters largely to supply chain management as a whole, we concentrate on protecting the privacy and preventing

[1] Singulation is defined as "Identifying an individual tag in a multiple-tag environment" by EPC Global Inc. [2]

clandestine tracking of the RFID tagged goods in transit. We design a scheme to dynamically change the RFID tag identifier and maintain information in a secure and distributed manner. We ensure that an authorized reader can have access to information about the tag, at any point in transit. We implement a prototype of this scheme using passive RFID tags and an RFID Reader system.

The rest of the chapters are organized as follows: Chapter 2 gives the background on RFID technology. We detail the concerns about RFID security and vulnerability in Chapter 3, and present the existing solutions to these problems in Chapter 4. We present our approach in Chapter 5 followed by the design detail in Chapter 6. We describe the prototype we have implemented in Chapter 7. We perform a security analysis of our proposed approach in Chapter 8. We describe the future work in Chapter 9 and conclude our proposal in Chapter 10.

Chapter 2

BACKGROUND ON RFID

In this chapter, we give a brief description of the RFID system, in terms of its components and communication methods.

2.1 Definition of RFID

RFID stands for Radio Frequency IDentification. This means the identification of objects using radio frequencies. Unlike bar codes, items tagged by RFIDs have unique identification. e.g. While the barcodes on two boxes of Sharpie markers would bear the same bar code, if they were RFID tagged, they would bear different and unique tag identifiers. Similarly, the popularity of RFIDs is also attributed to its non-line of sight operation.

2.2 RFID System components

An RFID system is composed of RFID tags and RFID readers.

2.2.1 RFID Tags

RFID tags are the elements of the RFID system that can be used to tag various items. They consist of a (antenna) coil and a chip/integrated circuit with some form of encapsulation. RFID tags can be classified into two types passive and active [12] based on whether

they are battery powered or not. Passive tags have no battery sources, and work on the energy supplied by the RFID reader, while Active tags have batteries as a source of energy. Hence they can transmit data over longer distances, although they have a lesser lifespan compared to the passive counterparts. It may be noted that the computational capabilities of Active tags are hence better compared to passive Tags. Semi-Active [13] tags are those which have on-board battery sources, but use them to power the tag circuits, rather than to power the communication with the reader.

RFID tags can also be classified depending on whether they allow writes to them or not [14]. Read-only tags permit the data on the tags to be only read and not written. Such read-only tags typically carry tag identifiers and not additional data on them. Write-Once-Read-Many (WORM) tags can be written once and read many times. In contrast, the read-only tags have their tag data written at time of manufacture only. Read-write type tags can be rewritten multiple times.

RFID tags may be able to carry more information than just their identifier. EPC Class 1 generation 2 [11] for example identifies tag memory as being divided into memory to store passwords for the tag, the actual identifier, an identifier defining the tag capabilities and memory to store user data. EPC Global defines the tag identifier [15], referred to as GID (General Identifier) to be composed of General Manager Number, Object Class and Serial Number. The GID 96 is latest version of the identifier and it also includes a unique serial number for each tag, and is shown in Table 2.1.

Table 2.1. EPC General Identifier (GID-96) format

Header	General Manager Number	Object Class	Serial Number
8 bit	28 bit	24 bit	36 bit

2.2.2 RFID Reader

RFID readers are the devices that are used to read from (and possibly write to) RFID tags. RFID readers can be classified as hand-held or fixed readers depending their use, either mobile or fixed. Handheld readers have a shorter read range compared to the fixed readers.

Readers follow different protocols like ISO [16] and EPC Global specifications [2]. Readers operate at different radio frequency ranges. The current versions of respective standards are EPC Class 1 and ISO 18000 standards. RFID Radio Frequency ranges are: low frequency (LF) which is around 128 KHz, High Frequency(HF) at 13.56 MHz and the Ultra High Frequency (UHF) at about 900 MHz and Microwave Frequency Range at 2.45 GHz to 5.8 GHz [17]. EPC Global's EPC Class 1 UHF Generation 2 standards are most current version of EPC UHF standards. The most commonly used frequency ranges are 13.56 MHz and 860-960 MHz [18].

2.3 Physical Communication between Reader and Tags

We mention the physical aspects of the communication between the RFID readers and tags for the sake of completeness. Communication between the reader and tags occurs via Magnetic Induction or by using Electromagnetic (EM) wave capture.

Magnetic Induction is similar to the transformer inductive coupling. This uses Faraday's principle of magnetic induction. An alternating current through the coil of a RFID reader causes an alternating voltage to appear across a coil in the RFID tag. This is also referred to as Near-Field Communication (NFC), and uses load modulation for communication, and is normally used for radio Frequency ranges under 100 MHz [12].

Electromagnetic wave capture, on the other hand uses the principle of capturing a Electromagnetic wave emitted by the dipole at the reader, by a smaller dipole at the tag [12]. The tags are beyond the near field communication range of the reader, hence back scatter

is used as the means of communication instead of load modulation. The communication takes place by changing the impedance at the tag. This is normally used for frequencies above 100 MHz, e.g. in 900 MHz HF and in 2.45 GHz UHF RFID systems.

2.4 RFID Applications

One of the main applications of RFIDs in the Supply chain management (SCM) world, in which RFIDs are used to track items throughout from their manufacturing through sale and potentially even beyond sale. Besides the manufacturing industry, the shipping industry finds many useful applications of RFIDs in transporting parcels, and baggages, in particular due to the non-line of sight operation. Similarly, the US Transportation Security Administration (TSA) uses RFID to track air luggage using RFIDs. The pharmaceutical industry similarly is running pilot projects on RFID tagging pharmaceutical drugs and medical supplies. Similarly, RFIDs are also used to track livestock. The applications of RFIDs are hence varied.

Chapter 3

RFID SECURITY AND PRIVACY CONCERNS

In this chapter, we look at the various concerns surrounding RFID security and privacy.

3.1 Threat model for RFIDs

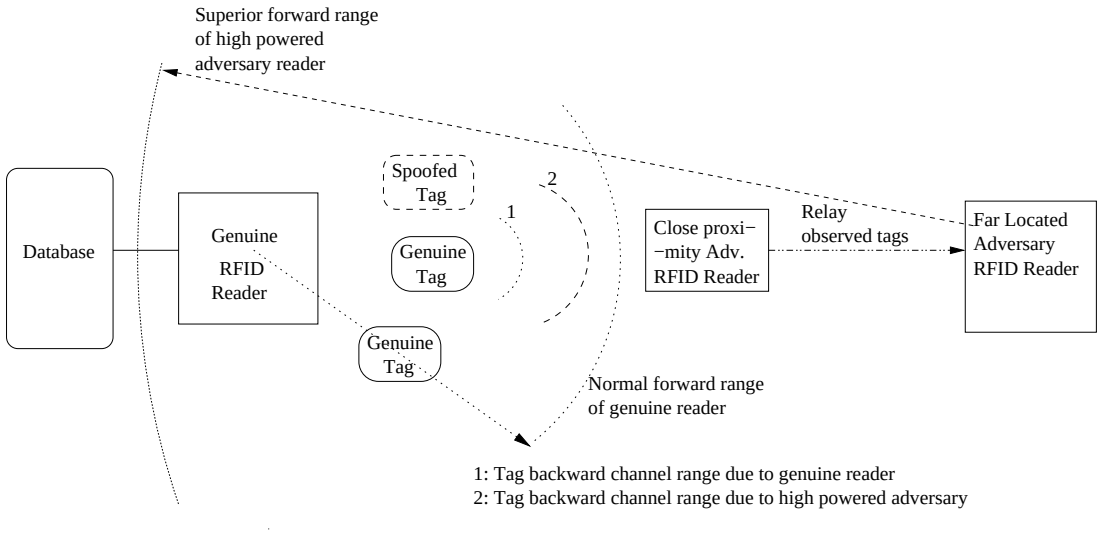

FIG. 3.1. A threat model showing spoofed tags and collaboration between adversary readers located at different distances

Figure 3.1 is a scenario which one would typically consider with respect to privacy and security for RFID. The ranges in the figure are for purpose of understanding, and are approximate. The spoofed tags shown in the figure lie within the read range of the

reader, hence would be read by the genuine RFID reader along with other genuine tags. The genuine RFID reader is shown connected to the back-end database system, which can store tag and reader related information. The close proximity adversary reader can listen to the backward (tag-to-reader) channel, while the far located adversary reader can eavesdrop only on the forward (reader-to-tag) channel. The high powered far located adversary causes the tag to reply on a backward channel which may be stronger (shown by "2" in Fig 3.1) than its normal range (shown by "1" in Fig 3.1). This stronger backward channel can be read by close located adversary and can be communicated to the far located adversary. A detailed description of an attack on similar lines is described in [19].

We identify the security vulnerabilities in the RFID system as follows:

- Attack on the singulation protocol between the reader and the tags
- Attack on communication between the reader and the tags
- Side-channel attacks

We consider these vulnerabilities in detail.

3.1.1 Attack on the singulation protocol between the reader and the tags

When a number of tags are present, and a reader wishes to communicate with the tags, the reader needs to follow a protocol to identify a unique tag to which it would communicate at a time. This process of choosing a unique tag is also called "singulation" of the tag. The protocol for singulation is subject to attack. Anti-collision algorithms are hence deployed to prevent collisions between the answers received from the tag during the process of the reader "interrogating" the tag. In a simple binary tree walking anti-collision algorithm, a reader would broadcast each bit of the tag's identifier (ID) to singulate it on the channel from the reader to the tag. A reader queries for each bit of the tag identifier. In response to a query from the reader, a tag will respond with a 0 or 1 depending on its bit value. If the

reader hears a common bit (either 0 or 1, not both) in reply from the tags for the bit, there is no collision. If a collision exists, the reader will recurse in the tree till it finds the unique tag. An adversary RFID reader in proximity could eavesdrop on this communication between the reader and the tag to learn the unique identifier.

3.1.2 Attack on communication between the reader and the tags

As shown in Figure 3.1, adversary RFID readers can be present in the vicinity. RFID communication ranges from reader-to-tag (forward) and tag-to-reader (backward) are asymmetrical. Hence, an adversary that is far located cannot snoop on the backward channel, but can can eavesdrop the forward channel. A close located adversary, on the other hand, can snoop on both channels.

RFID tags have a unique identifier. An adversary tries to snoop on the communication between the RFID readers and tags to learn their unique identifier. An adversary may clandestinely record the communication exchange to learn the unique identifier, by listening to either of the communication channels. This unique identifier typically is used as a key in a database which holds details of the items and has access controls. This problem is more severe for identity EPC class 1 [11] RFID tags, as a basic interrogation of a RFID tag is sufficient to learn the unique identifier.

An adversary could have two types of attacks on this system - active and passive. In a passive model, a malicious reader could simply monitor and silently record all the communication going on between the reader and the tag. It could create a database of this information. Such information could be sensitive or personal information and might be siphoned off to competitors. Such information can include information about the item, like its model, make, manufacturer and even the price. Similarly, in case of an item like a library book, it may have details of the patron to whom the book was issued. An active attack aims at disrupting the communication between the reader and the tag, which could

involve replay attack or modifying the recorded communication and then replaying it.

3.1.3 Side-channel attacks

This covers a variety of attacks [20], in which the adversary analyzes the behavior of devices in order to learn about the device. These attacks could involve use of timing attacks, in which an adversary measures the time taken by a tag to respond to a valid/invalid request and try to learn details about the tag from the response time. Similarly, a power analysis attack involves an intruder analyzing the amount of energy spent by the tag while doing a computation, like checking the password it has been supplied. Similarly, there are attacks which make use of the "radio finger-prints" of the RFID devices [18]. Avoine et al. in [18] describe "traceability" of RFID to be an issue at all different levels, from the physical layer to the application layer.

3.2 Privacy Aspects

Privacy can be defined as protecting the identity of the RFID tag (bearer). With RFID tags being used in daily use commodities like garments [5], in grocery stores, in shipped items [6] and also for medicines, it may be possible to track an individual based on the RFID tagged items she carries. RFID readers are easily available, for example, in cell phones [4]. An intruder has unfettered access to data on the RFID tag. Juels in [7] identifies these problems as clandestine tracking and clandestine inventorying. Clandestine inventorying is the ability of an unauthorized reader to access an item's inventory using the unique identifier on the tag. This problem is more elevated if the tag itself bears more information about the item, e.g. the price of the item. This problem can be overcome by storing an identifier on the tag, which is different from the database identifier for the tag, and not maintaining inventory information like the price on the tag. The problem of clandestine tracking is the ability of a rogue reader to track an item (or the item bearer) based the tag's

unique identifier. The adversary can correlate the time and place of sighting the RFID tags and infer details about the tagged item and hence the RFID tag bearer. This problem persists so long as the tag identifier is static. The problem is aggravated in case of RFID tags which allow a rewrite to the tag. An intruder may swap the genuine tag identifiers of two tags, which is referred to as a swapping attack.

Chapter 4

CURRENT SOLUTIONS

In this chapter, we explain the current methods adopted for overcoming the security an privacy aspects highlighted in the previous chapter.

The most preliminary solution possible is to “kill” a tag after it has satisfied its intended purpose. For example, in supply chain management, at check out, an RFID reader can be commanded to “kill” the tag on the product being sold. However, this would render the tag totally non-functional and this voids many post-sale benefits of uniquely identifying the goods to the store. Similarly, killing tags is not a solution for tags in E-Z Pass [21] or bank notes or sensitive documents like passports. We henceforth use the word tag to mean RFID tags, and similarly reader to refer to an RFID reader.

4.1 Toward solving security concerns

We look at the various methods been proposed with respect to the RFID reader-tag communication as well as regarding the singulation protocol, which helps to uniquely identify a tag in a group of tags.

4.1.1 Securing the singulation protocol

Weis, in his Masters thesis [22] speaks of “Silent Tree Walk” protocol. The threat model considers only far located adversary RFID readers, which can eavesdrop only on the

forward channel (reader-to-tag). Tags often share a common prefix, like the manufacturer identifier in the EPC code. The tags echo this common prefix on the backward channel (tag-to-reader) to the reader. The far located adversary reader cannot eavesdrop on this backward channel, hence does not know the common prefix. On detecting a collision, the reader sends out the bit padded by the common prefix on the forward channel in order to singulate the tag. Thus an eavesdropping adversary cannot hear the tag identifier. Weis suggests another tree walking scheme called Randomized Tree Walk in [22]. In this scheme, each tag generates a temporary random pseudo-identifier to serve as the tag identifier when the tag is being singulated. The tag produces a new pseudo-identifier each time the tag is being singulated. The reader will singulate the tag using a binary tree walk on this pseudo-identifier. At the end of the singulation, the tag replies with the actual identifier on the backward channel. This prevents am adversary from learning the tag identifier. Although this scheme does not need any shared secrets, one has to account for random pseudo-identifier generation by tags and collisions in the random ID generated.

Molnar et al. describe a “Private Authentication” scheme in [23]. In this scheme, a reader and a tag can authenticate to each using a shared secret, without revealing the secret to an eavesdropping entity. This scheme assumes both the reader and the tag to be possessing Pseudo Random Function (PRF) capabilities. The output from the PRF is used as a pad for communication. When the reader receives a value from the tag, it performs a lookup of the received value based on the secret of the tag in order to identify the tag uniquely.

4.1.2 Securing communication between the reader and the tags

With respect to the security of the reader-tag communication, various methods for protecting tag identity have been proposed as discussed below. We look at the most light weight solution from cryptography [9], or hashing [10]. EPC Class 1 Generation 2 [11]

requires the use of one-time pads for communication with the tags. Public cryptography involving re-encryption using a single public-private key pair [24] and Universal re-encryption [25] are some of the methods that have been suggested for Identity Class 1 tags.

Tags which are capable of computation, including active tags can implement more complex methods. Feldhofer et al. in [9] provide a solution that is based purely on encryption using keys. They classify a system as "open" for systems having unknown devices communicate with the tag and recommend public-private key techniques for these systems; while for "closed" systems in which a given set of readers could interrogate the tag, they recommend a shared secret key approach. Similarly, approaches which use a one-way hashing based on a shared secret between the reader and the tag have been proposed. Variants of the challenge-response protocol have been suggested using one-way or two-way challenge response methods.

4.2 Toward protecting the privacy of tags

4.2.1 Toward uniquely identifying a given tag

EPC Global Inc. [2] provides a mechanism to weave in all the RFID tagged items into a single integrated world with the EPC Global Network [3]. It has core components for identifying goods called the ONS (Object Name Servers), and a service to provide the information as needed called the EPC IS (EPC Information service) [3], and authentication being provided by EPC Data Service [3]. Although the EPC Network is able to identify the goods uniquely, it does not provide explicit mechanisms for protecting goods in transit from being clandestinely tracked.

4.2.2 Using another device to protect tag identity

Juels et al. in [26] suggest the notion of using a "tag-blocker" device to protect the privacy of the consumer for the tagged items she has purchased. The ultimate purpose of the tag blocker approach is to prevent the reading of tags by readers, when the consumer wants privacy and does not want to reveal what she purchased. However, since the tag itself is not "killed", it can respond to requests when needed. This scheme exploits the basic tree-walking protocol used by the reader to singulate the tag. The "Universal blocker" simulates the range of all possible tags in the tag identifier space. When a reader is querying the tags to determine one bit of its identifier, the tag blocker would simply reply with both a 0 and a 1, thus causing a collision in the reply. A reader would assume it detected the collision, and would start recursing into the tree, just to face more collisions, thus preventing any reads. A "Selective blocker" similarly would simulate a selective range of tags, particularly, zones which are marked as privacy zones. The tag blocker would not interfere with the reader till the reader enters the privacy zone.

Other methods of blocking reads include use of a "Faraday cage" which is a device used to shield the tag from radio frequency (RF) signals. The RF signals cannot enter the cage and hence cannot detect the item placed in the cage. This approach however, would need different shielding containers to block out different ranges of frequencies. This Faraday cage, which is a a foil-lined bag works well for objects that can fit into the bag. This can be subject to misuse to steal items without them being scanned. However, this concept would not work for items like garments, which the consumer would be wearing. "Active Jamming" is a method by which a consumer pro-actively intends to protect her privacy by using a powered device to generate a signal that disrupts the RF interaction between the RF tag and the transponder. This signal may be a simply a radio signal which is broadcast. This approach can be easily detected, if not penalized for causing disruption to the reader, and possibly other devices in the frequency.

4.2.3 Protocol based methods to protect tag identity

The approaches described in Section 4.1.1 in which the reader and tag engage in secured protocol to authenticate to each other without letting an eavesdropper learn any of the secrets contribute to protection of tag identity. Juels in [27] suggests a tag maintaining multiple pseudonyms and each replying with a different tag pseudonym on each tag read. These pseudonyms are released at a controlled rate and an authorized reader can supply new pseudonyms to the tag. However, this would require custom modification to the commonly available passive tags. Similarly methods like challenge-response as mentioned in Section 4.1.2 are used to protect the identities of tags.

Chapter 5

PROPOSED APPROACH

5.1 Design Objectives and Focus

The aim of our scheme is to protect the identity of tags, and prevent them from unauthorized access, without hampering their visibility to authorized readers. Our design objectives include:

1. To prevent the RFID tag identifier from unauthorized access
2. To ensure integrity of the RFID tag identifier
3. To provision for tracking at any point in transit
4. To provide different levels of access to information about a RFID tag
5. To provide flexibility to dispatcher of goods to choose a privacy policy for tags (and possibly update it)

5.2 Overview of the approach

Using a local entity and an Authenticating Agent, we provide the access control which would otherwise require custom manufactured expensive tags. We note that our scheme is suited for applications like tracking of shipments, and we consider a threat model where

an adversary seeks to clandestinely track the items in transit, but assume physical security of the truck and its contents. In the security enhanced variant of our approach using active tags, we can provision more features like tamper-detection, tag independently alerting the Authenticating Agent, while retaining the key aspects of our original design. We detail the actual entities and their characteristics in Section 6.1.

5.3 Assumptions

We are concerned with the security of the RFID communication and hence assume secure channels of communication (like wireless links/local communication) in the scenarios we describe. Similarly, we assume the physical security of the devices involved, like an intruder not being able to have physical access to the contents of the truck. We assume that the RFID tag reprogramming events are timed so as not to interfere with the read attempts of authorized RFID readers.

5.4 Privacy and Security Aspects

We note that the problem of clandestine tracking and clandestine inventorying manifests so long as the tag identifiers are static. Similarly, attacks such as traceability, hacking the password of a tag [20] and similar side-channel attacks are outside the scope of our solution. The focus of our approach is on privacy, but we need to secure the reader-tag communication. We hence choose the most light-weight yet secure method which lets us individually encode/decode tags. Further, our scheme involves a large number of keys to be used on a temporal basis as we re-encrypt the original tag identifier with a different key (or key pairs) over a period of time.

5.5 Feasibility Study

The requirements of our system in terms of RFID can be listed as follows:

1. Ability to uniquely identify a single tag

2. Ability to rewrite a particular tag with a new identifier

3. Authorized write mechanism

We explain the feasibility considerations for all the above scenarios. Uniquely identifying a single tag is also known as singulation. Variations of a binary tree walking protocol like Silent Binary-Tree Walking [10] or Randomized PRF Tree Walking Algorithm [28] can be used to singulate a tag in addition to any proprietary protocol implemented by the RFID vendors. In order to explain the feasibility aspects, we compare the solutions offered by two leading vendors of passive RFID systems. A rewrite of a particular tag with a new identifier is possible in EPC Class 1 Generation 2 [11] (referred to as Generation 2 henceforth). In implementations prior to Generation 2, some vendors require that a single tag be present. One vendor allows to selectively erase a tag in a group of tags, in which the tag is singulated and then its EPC code is erased, while another vendor allows selectively writing additional data to a tag with a specified EPC code in a group of tags. For the authorized write mechanism, we note that EPC Class 1 Generation 2 [11] provides access controls in form of a password to modify the tag's data. For implementations prior to Generation 2, we could either have custom modification to the tag by which it allows a write only on being supplied the password, or we alternatively design a work-around for the same. Some vendors implement variants of this in prior to the Generation 2 specifications. One vendor offers support for locking tags with a specific password, which means that the data on the tag cannot be replaced, until the tags are reprogrammed (or killed[1]) using the password to reset its data and subsequently writing new identifier onto the tag.

The following factors contribute to the timing characteristics of the proposed system:

1. Method of encoding tag identifiers

[1]Some vendors prior to Generation 2 erase the tag memory, but permit subsequent tag reuse on "killing" a tag.

2. Tag reprogramming

3. Communication delays

Depending on the method chosen for encoding tag identifiers, like encryption (including symmetric encryption or asymmetric encryption) or hashing (which involves lookup for values in a hash table), the complexity and consequently the time required for calculating the new tag identifier changes. Besides the computing overhead, the time required for an actual write to the tag needs to be accounted for. This includes the time required by an RFID reader to singulate a particular tag to write to it. Our preliminary experiments we noticed an increase in write time for a single tag from 100 ms to order of 500 ms when a group of tags is present. We attribute this to singulation protocol followed by the device. We consider re-programming of tags be feasible, so long as we limit the number of tags in the transit vehicle. Communication delays exist in terms of access time via the communication media. Such communication with a central server is possible as exemplified by GE VeriWise Asset Intelligence system [29]. Further, our protocol has a security parameter that defines the length of the encryption key(s) and the time interval for re-program of a tag. We are currently working on developing a prototype for our system.

Chapter 6

DESIGN

6.1 Entities

Our scenario involves the following entities - the Sentinel, the Authenticating Agent, the Interrogator and the Dispatcher. As shown in Fig 6.1, the Sentinel and Authenticating Agent share the responsibility of protecting the identity of RFID tags. An Interrogator may be a genuine entity like a customs officer, or may be a thief planning to steal expensive items off the truck, hence wants to know the identity of tags. The Dispatcher is an entity which dispatches goods to the destination. The roles of the entities are mentioned in Table 6.1. We note that our Authenticating Agent, unlike the Mobile Agents mentioned in [30] is located at the transit headquarters and is not mobile. Rather, it is an authentication server which also implements a policy based access control mechanism.

Table 6.1. The roles and capabilities of the interacting entities

Entity	Role	Connectivity	Location
Sentinel	RFID reader	Internet and Local	Transit truck
Auth. Agent	Authentication	Internet only	Fixed
Interrogator	RFID reader	Internet and/or Local	Vicinity of truck
Dispatcher	Policy Rules	n/a	n/a

We note that the Sentinel is required to be in the Transit truck for the approach involving passive tags. Similarly, the Interrogator should be in vicinity of the Transit truck

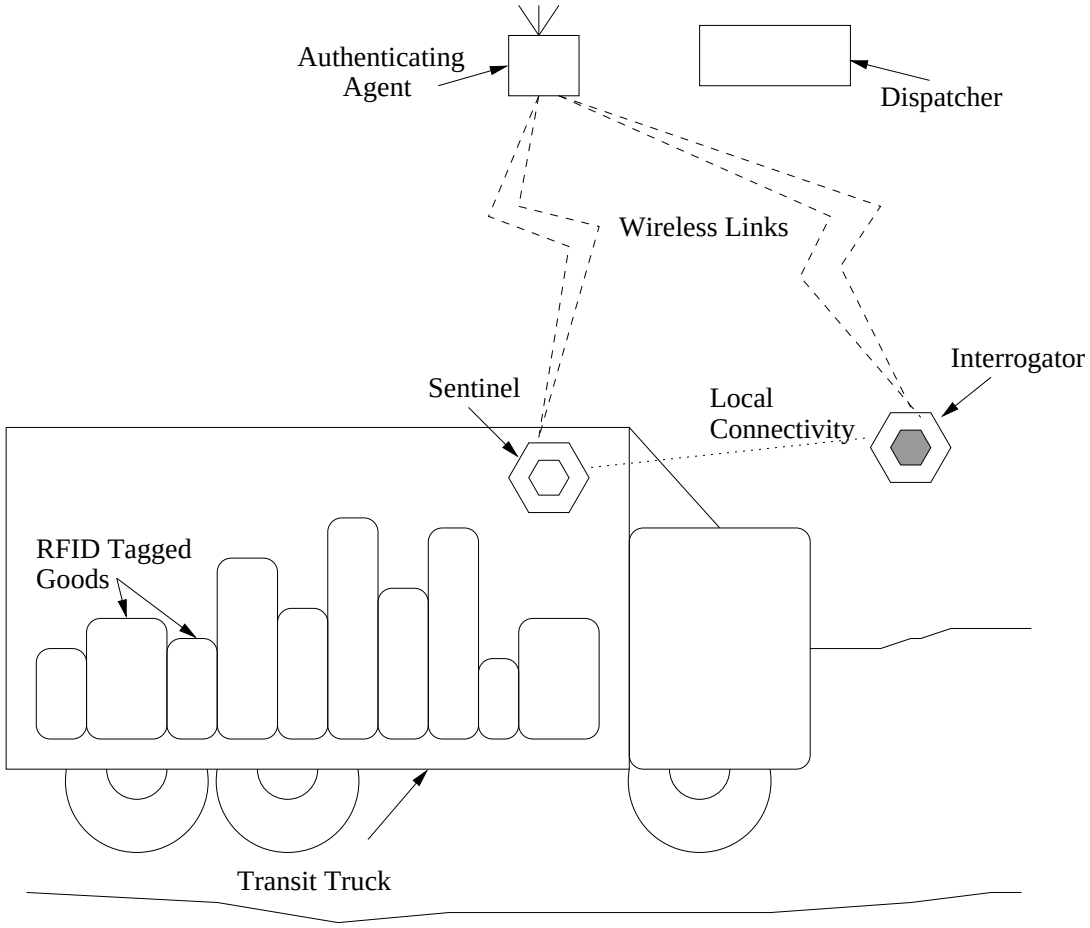

FIG. 6.1. Authenticating. Agent and Sentinel collaborate to give an authorized Interrogator access to tag identifiers

for the passive tags approach. When using active tags, we could have Sentinels located as far as 100 meters from the transit truck. Hence it could be possible to have Sentinels (like cellphone towers) deployed at different spots along the route of the transit truck.

6.2 Overview of Interaction

The steps an Interrogator follows to learn the identity of the RFID tags being carried are illustrated in Fig 6.2. As the first step in the interactions, the Interrogator reads the encoded RFID tags, but cannot understand the tag identity, hence communicates with the Authenticating Agent and gets cookies for the tags. The Sentinel issues it decrypting keys using which the original tag identifier can be computed. The Authenticating Agent can give more information about this identifier.

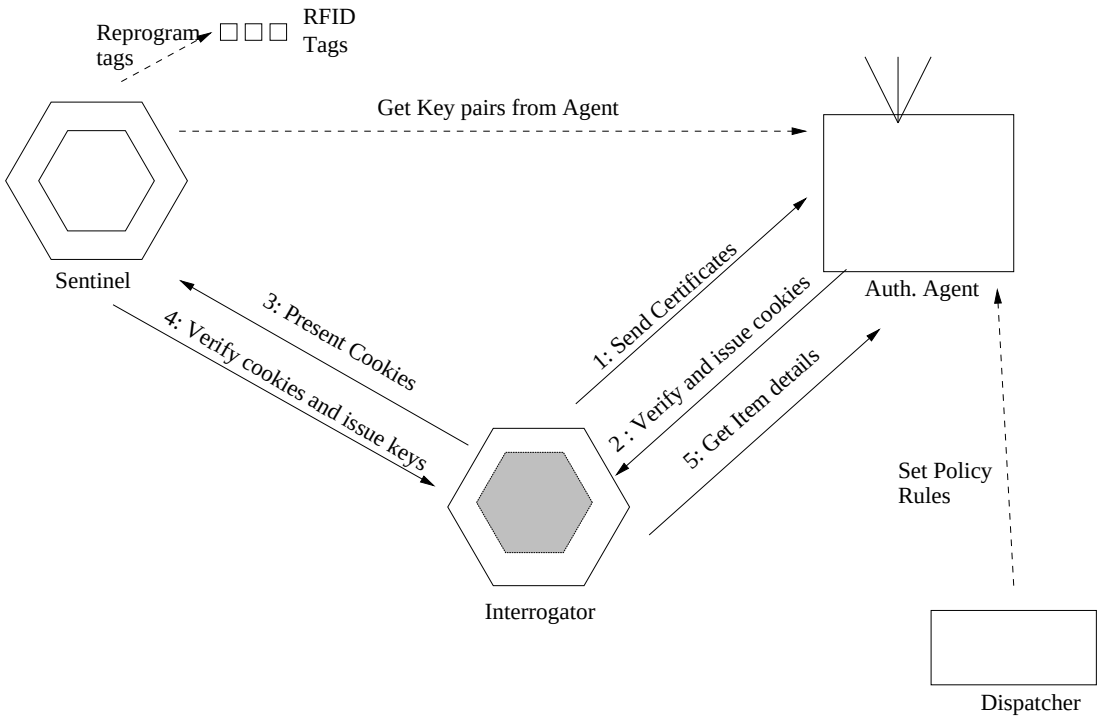

FIG. 6.2. Interrogator communication with the Authenticating Agent and Sentinel to learn the tag identifiers

6.3 The Protocol

Our protocol has encryption at tag level. We are not bound to any specific implementation; we can choose from symmetric/asymmetric encryption, or one way hashing of tags scheme to use the most light-weight yet secure method which lets us individually encode/decode tags. The Authenticating Agent has details of the tags, including their original identifiers and their respective passwords. We note that if we use cryptography in our approach, a tag could hold the digital signature of the Sentinel in the tag identifier or the tag data or a combination of the two. This would guarantee authenticity of the tags.

6.3.1 Approach using passive tags

We describe the steps by which the interaction between the entities occur:

- The Sentinel, from time to time (depending on security level) connects to the Authenticating Agent. The Authenticating Agent has (or generates) a large number of keys, and assigns a different set of keys to a requesting Sentinel.
- The Sentinel reprograms the tags in the truck, at repeated time intervals depending upon the security level.
- The Interrogator is in proximity of the tags and reads them. However, the Interrogator cannot identify them as they have been encrypted by the Sentinel. It continues by the following steps to get access to the tag details:

    1. The Interrogator is directed to the Authenticating Agent. The IP address of the Authenticating Agent can either be pre-programmed or could be learned from the Sentinel via local connectivity. The communication between the Interrogator (or Sentinel as a proxy) and Authenticating Agent occurs over a secure channel. The Interrogator supplies its credentials and information about tags it

wishes to identify. After verifying certificates and checking Interrogator access privileges for the tags, the Authenticating Agent issues access to the Interrogator in the form of cookies.

2. The Interrogator presents these cookies to a Sentinel, which issues it the decryption keys and encrypted identifier for the tags. It may be noted that using this encrypted identifier, the Interrogator can now track the tag until the Sentinel reprograms the tag. The Interrogator uses the decryption keys on the tag to get the actual tag identifier.

3. The Interrogator then presents this identifier to Authenticating Agent, along with its certificates. The Authenticating Agent uses a policy based engine (with policy rules chosen by the Dispatcher) to decide on level of access to information about the tag.

6.3.2 Approach using active tags

The proposed approach for a solution using active tags is almost the same as the solution using passive tags, except for the read ranges and security enhancements in the active tags. Fig 6.3 illustrates the scenario of using active tags in place of passive tags. We enumerate the changes as follows:

1. The condition that the Sentinel needs to be co-located in the truck is relaxed as shown in Fig 6.3. It can be located at a distance from the truck; as mentioned above, Sentinels can be deployed similar to cellphone towers to communicate with the tags in truck in vicinity.

2. As the truck passes through the route, different Sentinels would be responsible for the truck. This situation is the same as a Sentinel handing off the tags to another Sentinel (via communication with the Authenticating Agent) in the passive tags solution.

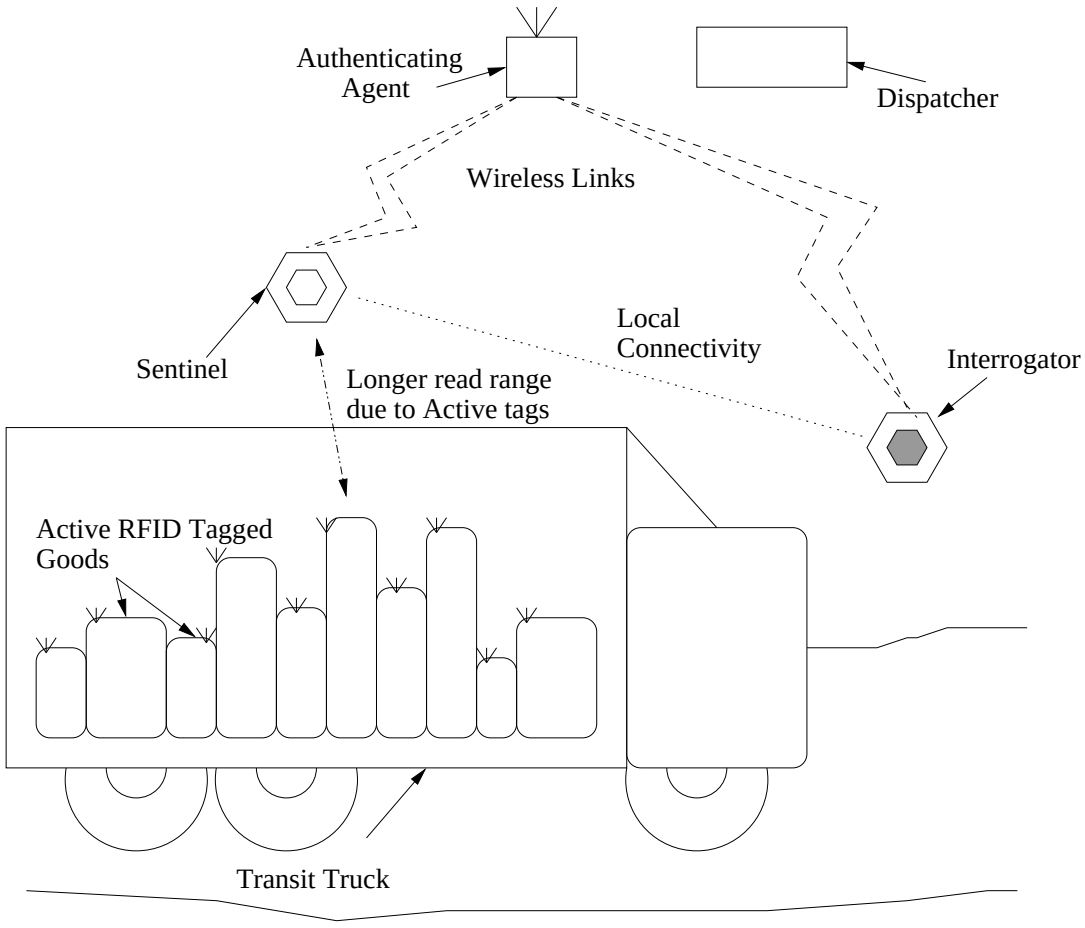

FIG. 6.3. In an approach using active tags, the tags have a longer communication range and have wireless connectivity

3. Similarly, instead of the Sentinel rewriting an identifier onto the set of tags in the truck, the Sentinel could connect to the tags and inform each tag of the new identifier it is supposed to use for communication with other entities.

4. We assume the active tags have wireless connectivity. Active tags can communicate at high speeds, hence a Sentinel as well as Interrogator could communicate with trucks moving at speeds even as fast as 100 mph [31]. The active tags could also have sensors on them, making the tags tamper-proof.

5. Similarly, in an event of detecting an event like high temperature or a possible attempt to tamper security, active tags could send out alerts to the Sentinel or the Authenticating Agent.

Chapter 7

IMPLEMENTATION DETAIL

We describe a working prototype of the protocol using passive RFID tags.

7.1 Prototype Detail

We implement our proposed approach for passive tags, using a computer system in addition to an unmodified RFID reader.

7.1.1 System Components

We experimented with the RFID hardware for our prototype as described in Table 7.1. The software components of our system are described in Table 7.2. We ran our experiments using a desktop computer with a Intel Pentium 4 Dual processor at 2.39 GHz, 1 GB RAM and running Microsoft Windows XP Professional operating system. We used Java version 1.4.2 for developing our software prototype.

Table 7.1. The Hardware components used for experiments

Name	Detail	Capabilities	Connectivity
Reader	Symbol AR400 RFID Reader	Read/Write RFID Class 0/1 Passive Tags	Ethernet
Tags	Alien ALL-9334 RFID Tags	EPC Class 1 Passive Tags (96 bit R/W)	n/a
Tags	Symbol Dual Dipole R/W Tags	EPC Class 1 Passive Tags (256 bit R/W)	n/a
Tags	Symbol Single Dipole R/W Tags	EPC Class 1 Passive Tags (256 bit R/W)	n/a

7.1.2 Prototype Design

Our proposed system has the following components as mentioned in Table 6.1 on page 22 - The Authenticating Agent, the Sentinel with a number of RFID Tags, an Interrogator and the Dispatcher. However, this requires two RFID readers - one each for the Sentinel and the Interrogator. Further, the Sentinel and Interrogator need access to the Internet. We do not deal with actually porting capabilities on the actual RFID reader. Rather, we simulate the existence of the entities with the use of an RFID reader (without any modification to its firmware), RFID R/W Passive tags and a computer (as described in Section 7.1.1) which implements the protocol.

We hence build a prototype as shown in Fig 7.1, using a one RFID reader from Symbol Technologies AR400 [32] (with firmware prior to Generation 2 of EPC Class 1 standards) and Alien ALL-9334 tags [33]. The Sentinel and Interrogator are software entities, which have access to the RFID reader. Instead of having two separate RFID readers, we use a single RFID reader for the interaction with RFID tags. Similarly, the Authenticating Agent and Dispatcher are software entities, which can be reached via the Ethernet. The interaction among all the components happens over sockets over the Ethernet.

7.1.3 Prototype Capabilities and Limitations

The interaction among all the components happens over sockets over the Ethernet, we detail the interaction as follows: The Prototype Capabilities are as listed in Table 7.3.

We list the limitations of our prototype:

1. Limited Tag Identifier rewrite capability: The RFID reader we are using is able to

Table 7.2. The Software components

Purpose	Name	Detail
Reader Communication	Symbol API	Java API to communicate with the Reader over the Ethernet
Prototype Software	Java 1.4.2	Java Socket API for communication between the Entities

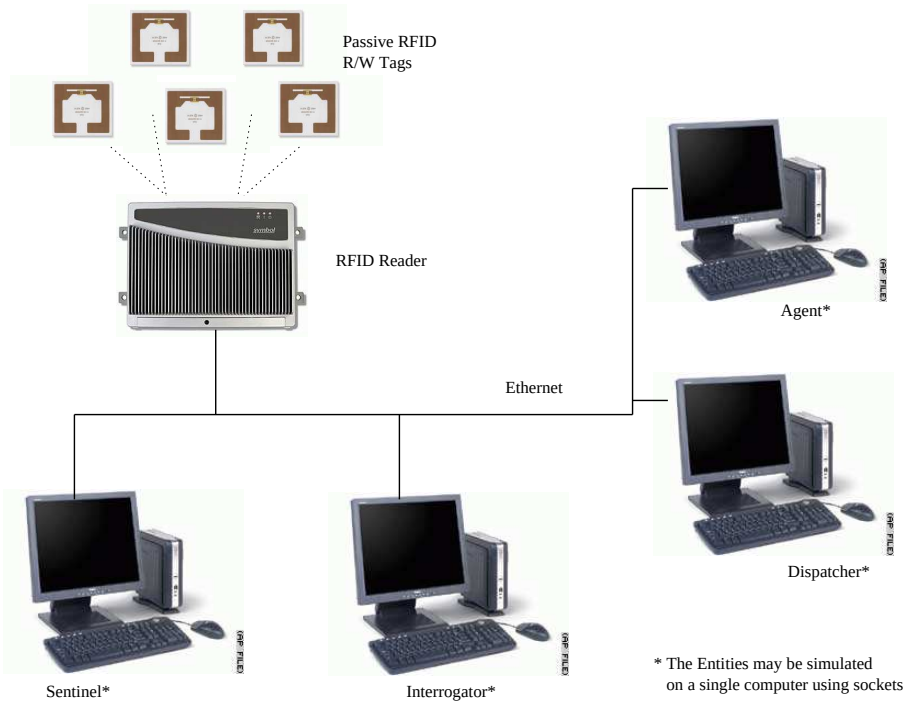

FIG. 7.1. Our prototype consists of a RFID reader connected to a computer system

Table 7.3. Prototype Capabilities

Capability	Implementation	Actual
Comm. between Auth. Agent and Sentinel	Sockets over the Ethernet	Wireless Comm.
Comm. between Auth. Agent and Interrogator	Sockets over the Ethernet	Wireless Comm.
Comm. between Sentinel and Interrogator	Sockets over the Ethernet	Local Comm.
RFID tag Read capability	API of the RFID reader	Same as implemented
RFID tag Write Identifier	API of the RFID reader	Same as implemented

rewrite the Tag identifier onto the tag. However, it is manufactured prior to Generation 2, and does not singulate the tag before it overwrites the Tag identifier. If more than one writable tag is present in the write range of the reader, all the tags will be overwritten with the same identifier. Hence, we need to shield other tags from getting overwritten. EPC Class 1 Generation 2 [11] mentions that upon singulation, the EPC data as well as the identifier can be overwritten hence solving the problem for the new Generation 2 compliant systems.

2. Limited Password Protected Mechanism: The RFID system we currently use is manufactured previous to generation 2 and does not provide for a "Lock" or "Access" mechanism by which we can prevent the tag from being accidentally overwritten or in fact protect it from an adversary till a genuine reader unlocks it using the password. Generation 2 of EPC Class 1 provisions for an Access passcode, without which the tag cannot be overwritten.

3. Use of Ethernet for communication: All entities communicate over the ethernet in place of their expected wireless links, or local connectivity. Since we provide for the prototype in Java, our prototype can be easily ported to any other implementation.

4. Accuracy of Read/Write: We encountered some inconsistencies in the number of tags being reported based upon the orientation and distances between the tags, which we mention in Section 7.3. Similarly, we found that the write operation did not have the desired overwrite effect on certain tags. In extreme cases, we found that a tag could get destroyed when a write is performed on it.

7.1.4 Programmable Parameters in the Prototype

1. Reprogram Interval: This defines the time interval after which the Sentinel reprograms the Tag identifiers on the tagged goods.

2. Communication Security Parameter: This defines the strength of encryption used for the communication between the interacting entities.

3. Simple Policy Mechanism: This defines the policy by which the Authenticating Agent gives access to an Interrogator for the tags.

7.2 Protocol Detail

The interaction between the entities proceeds as described in the Fig 6.2. The entities exist as software entities. The Sentinel and Interrogator get their tag reads from the RFID reader shown in Fig 7.1. We describe the details of the entities in more detail:

- The Authenticating Agent maintains a list of all possible tags. For each tag, it stores information about the tag like meta data pertinent to the tag, which gives a description of the item, as chosen by the Dispatcher.

- The first level of authentication at the agent is to give access to the Interrogator based on the meta data it presents. This could have access control mechanisms, like using the certificate of the Interrogator at authenticate access. Besides authorized entities like company employees, a law enforcement officer or a customs officer would be granted access to the tagged items, in the form of a cookie.

- The second level of authentication decides how much detail about the tag is to be revealed to an Interrogator when it presents the decoded tag identifier for the tag. The Authenticating Agent can have policies by which it decides to give information about the tag to an Interrogator. An example of this could be to allow an employee of the dispatching company to query about the tag for basic information, while authorized users would get more details about the item. The Authenticating Agent can have a policy engine to determine the access detail. For our implementation, we use a simple

method for determining the access level, however it can be extended to incorporate a rule-based engine.

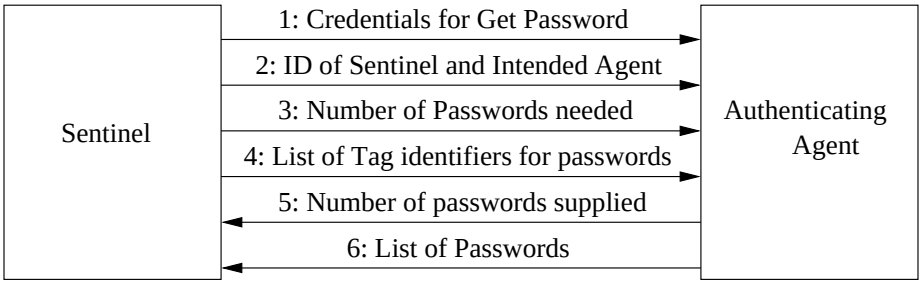

FIG. 7.2. Sentinel Interacts with Authenticating Agent to get passwords for tags

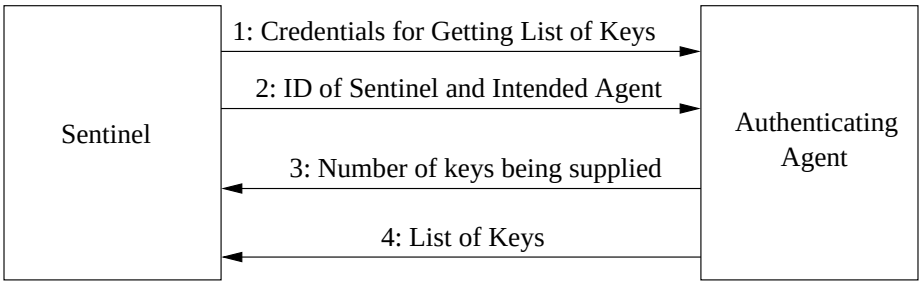

FIG. 7.3. Sentinel Interacts with Authenticating Agent to get list of Keys from Sentinel

We explain the interaction as follows:

- The Sentinel communicates with the Authenticating Agent to notify it of the tags present with it. In this process, the Authenticating Agent marks the respective tags as being present with the specific Sentinel and issues it passwords to write to the respective tags. This is explained in Fig 7.2.
- The Sentinel, depending on the Security parameter, connects to the Authenticating Agent to get a list of keys as shown in Fig 7.3. The Sentinel can pick any of the keys it has to encode each of the tags.

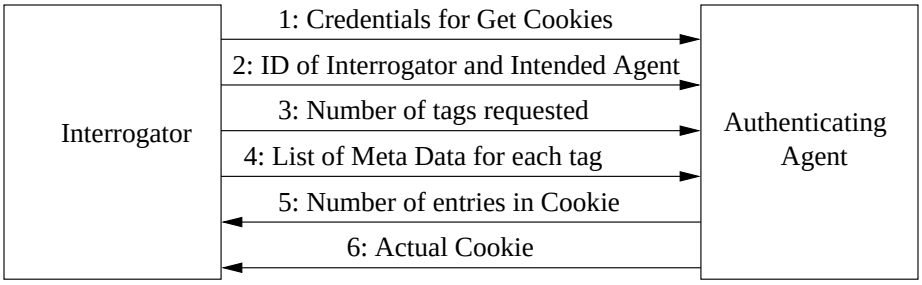

FIG. 7.4. Interrogator interacts with Authenticating Agent to get Cookies for tags

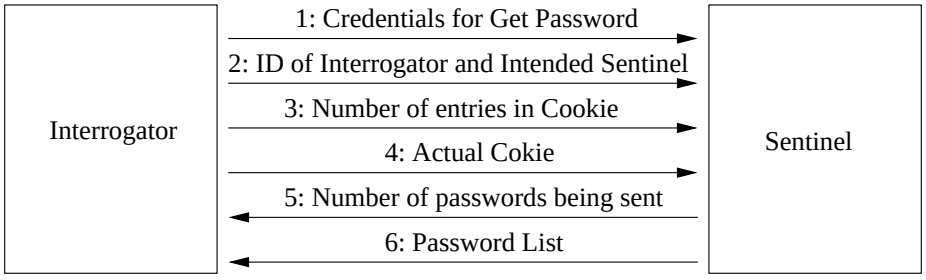

FIG. 7.5. Interrogator interacts with Sentinel to get passwords corresponding to Cookie from Authenticating Agent

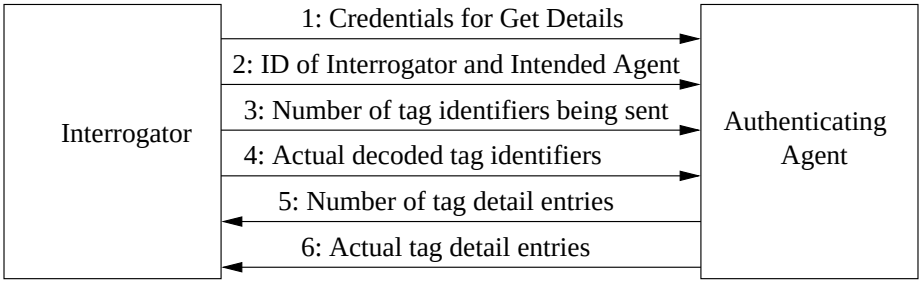

FIG. 7.6. Interrogator gets the details of the respective tags from the Authenticating Agent

- The Interrogator has read all the encoded tags and wishes to get the original identifiers for some tags, in order to verify the existence of the goods of interest onboard the truck. It follows the following three-step process to get the tag identifiers:

    1. The Interrogator connects to the Authenticating Agent to get a Cookie for the tag(s) it wishes to identify as shown in Fig 7.4.
    2. The Interrogator then presents this Cookie to the Sentinel to get the decoding keys for the tag(s) as illustrated in Fig 7.5.
    3. On decoding these tags, the Authenticating Agent, depending on the access details for the Interrogator will issue it details of the item as shown in Fig 7.6.

7.3 Results

7.3.1 Results from interactions with actual tags

We present our results from interactions with various tags mentioned in Table 7.1. For the actual readings, we used the Symbol AR400 reader and a number of Alien ALL9334 tags, in addition to tags from Symbol Technologies as mentioned in Section 7.1.1. Our focus in all the experiments we describe ahead is to determine the demonstrate the feasibility of our scheme. Our proposal does not focus on the study of physical aspects of RFID tags, like the orientation sensitivity, read/write range, read/write speeds and so on. Relevant performance benchmarking about Passive RFID tags is given by Ramakrishnan et al. in [34]. We wish to demonstrate that we were able to implement the scheme using off the shelf components like RFID readers and tags without any modifications to them. In reality, we could port the protocol to the reader system itself, but we do not focus on this aspect. We present the results of experiments using the Symbol AR400 reader and RFID tags. We note that the readings may be offset by objects in the surroundings like metallic objects reflecting the signals, as well as absorption of signals by fluids. However, we do not

explore this aspect and assume that the effect of these objects is uniform on all the readings we record and does not affect the readings to a large extent. We present the tag read, write and singulation timings for our system in this section.

- Time taken by the reader to read tags

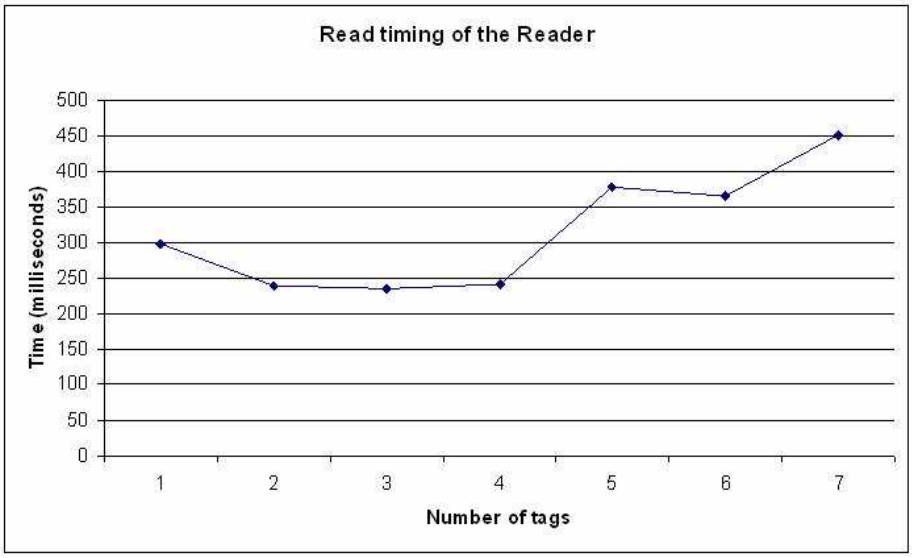

FIG. 7.7. Time needed by reader to read tags

We recorded sets of timings needed by the reader to get/write the tags. We observed that the tags performed best when the tags were placed perpendicular to the axis of the antenna with maximum surface and hence coil area being exposed to the Antenna. We present the time taken by the reader to read a different number of tags in Fig 7.7. We have plotted the average read time needed by the reader over a large number of reads for different number of tags. We observe that the average read time varies from 250 ms to 450 ms for upto 7 tags. We note that there is interference by metallic objects in the setup for the tags, hence it is possible that the reader is getting reads with low strength and is trying to determine if tags are actually present or it is noise in the readings. Further, there are collisions in the tag reads. The orientation of the

tags, and the distance between them also affects the reads.

- Accuracy of the reader in reading tags

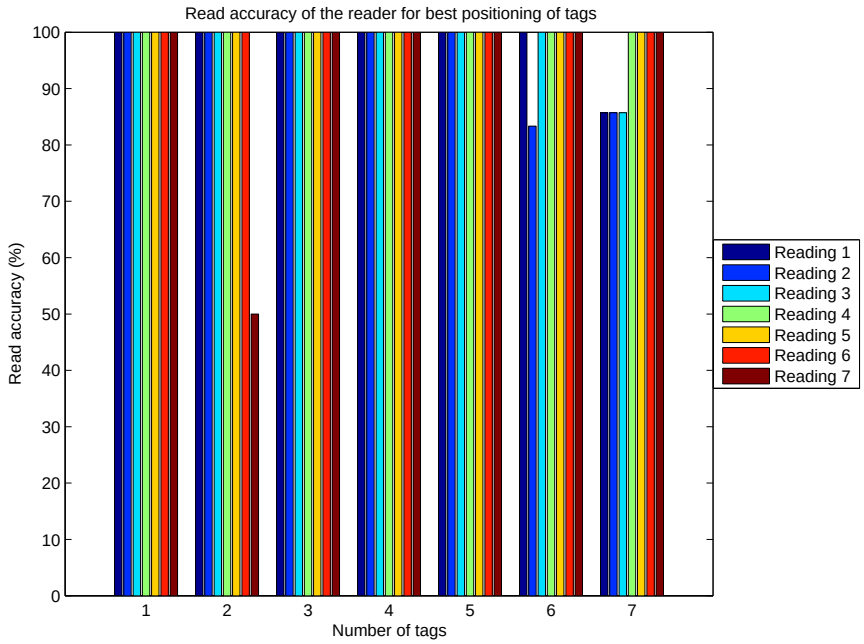

FIG. 7.8. Reader read accuracy for best positioning of tags

We present the percentage of number of tags read as against the number of tags actually present in our graphs pertaining to read accuracies. Fig 7.8 gives the accuracy with best positioning of tags, while Fig 7.9 gives the read accuracy with random positioning of tags. In both graphs, we plot the values of the read accuracies observed over multiple runs of the experiment with increasing number of tags. We get almost 100 percent accuracy with best positioning of tags, for random positioning we encounter lower accuracy values. Whereas in both graphs, the read accuracy is almost 100 percent when fewer tags are present, the read accuracy varies in different runs of the experiment for the graph with random positioning, since the positioning of the tags affects the read accuracy of the reader. We note that in random positioning, we

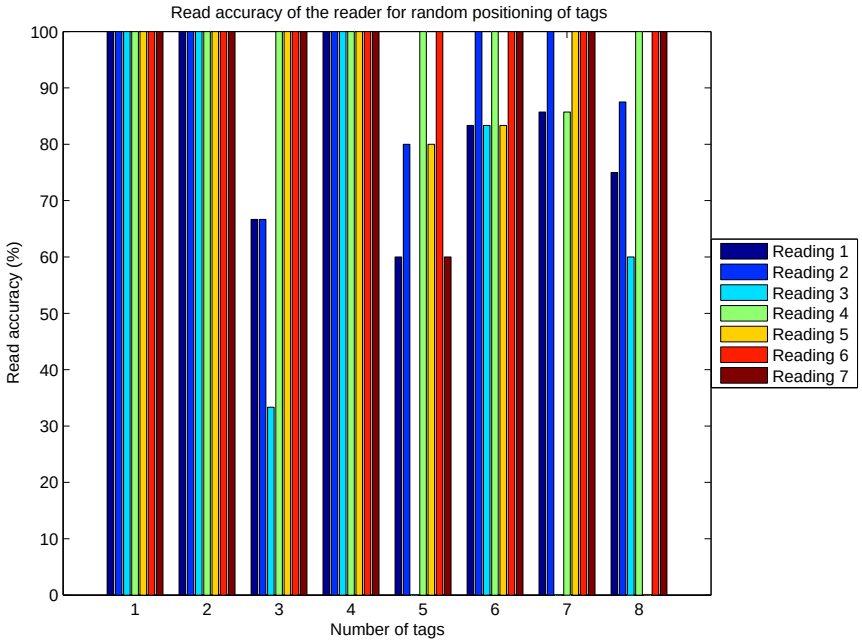

FIG. 7.9. Reader read accuracy for random positioning of tags

do not place tags along the null of the antenna (where the reads are zero), rather we place tags at different orientations, with varying distances from other tags.

- Time required by the reader to singulate tags

  We attempt to determine the time needed by a reader to singulate tags. We note that the Symbol reader we use has firmware prior to Generation 2. When instructed to rewrite the tag identifier (EPC code) on the tag, it will rewrite the identifier to all tags which are present in the write range. We note that the reliable read and write ranges experimentally observed by us were 5 feet at a maximum. We note that this is also specific to the tags we used, as for passive tags, more the size of the tag, the higher the read/write range to it. The Symbol reader however singulates tags when overwriting the additional data to a tag, that is, is able to identify a specific tag and can overwrite its data. Hence we performed experiments in which we wrote new

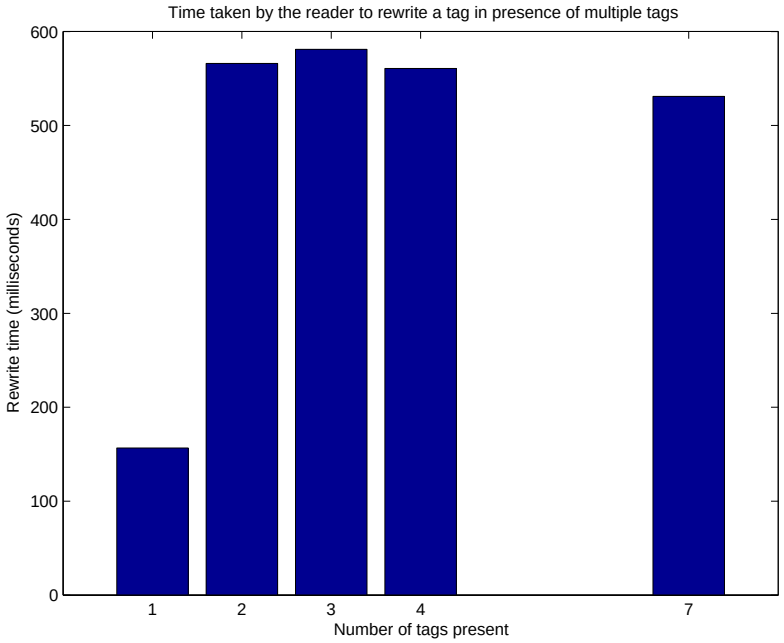

FIG. 7.10. Time taken by reader to rewrite to a specific tag in presence of a number of tags

data (not identifier) to a specific tag in the presence of different number of tags. The graph in Fig 7.10 shows the time taken to rewrite a tag with new data. We note that the tags from Symbol have additional data along with the EPC identifier. We increase the number of tags present, to note the time required by the reader to singulate the tag it wishes to overwrite. We observe that the least amount of time is taken by the reader when the tag whose data is to be overwritten is present by itself, without any other tags in vicinity. The reader took varying time to rewrite the tags in presence of multiple tags, which can be attributed to the proprietary singulation protocol followed by the reader to read the tags. The inconsistencies in the readings may be attributed to varying read accuracy as well as tag collisions depending upon the orientation of tags and the distances between them.

7.3.2 Results from simulations

We describe the results we get from the simulated interactions between the tags and readers. We note that since the reader was unable to re-write the tag identifier for a specific tag, we simulate the reading and writing of tags through software for the experimental setup described ahead. Parameters defined for the simulation include the number of tags, and a number of programmable intervals, including the time to reprogram tags, Interrogator query timings and so on. We can simulate a large umber of tags in the program, however, for the purpose of our readings, we have simulated up to 100 tags.

Table 7.4. Configurable parameters used in the simulations

Number of tags	Encryption Level	Sentinel Reprogram Interval	Sentinel get Keys Interval	Interrogator Query Interval
0 to 100	0 to 2	31000 ms	140000 ms	5000 ms

Table 7.4 defines the values of the parameters used in the simulations. We support different levels of encryption. The Sentinel reprogram time interval is the time interval after which the Sentinel will reprogram the tags. After a fixed amount of time, the Sentinel communicates with the Agent to get a new set of keys. the Agent issues a fixed number of keys to the Sentinel. We also define the time period after which the Interrogator will query the Agent and Sentinel for tag details.

We note that our implementation is in a high level language(Java), we do not have optimizations for performing the operations. At the same time, garbage collection and network delays may offset the readings in the experiments, but we assume that these factors do not have considerable effects on the readings. We next describe the results we have obtained from the interactions of the Interrogator with the different entities. We describe the results we obtain for the three steps followed by the Interrogator to get details of the tags of interest.

1. Interrogator gets Cookies from Authenticating Agent

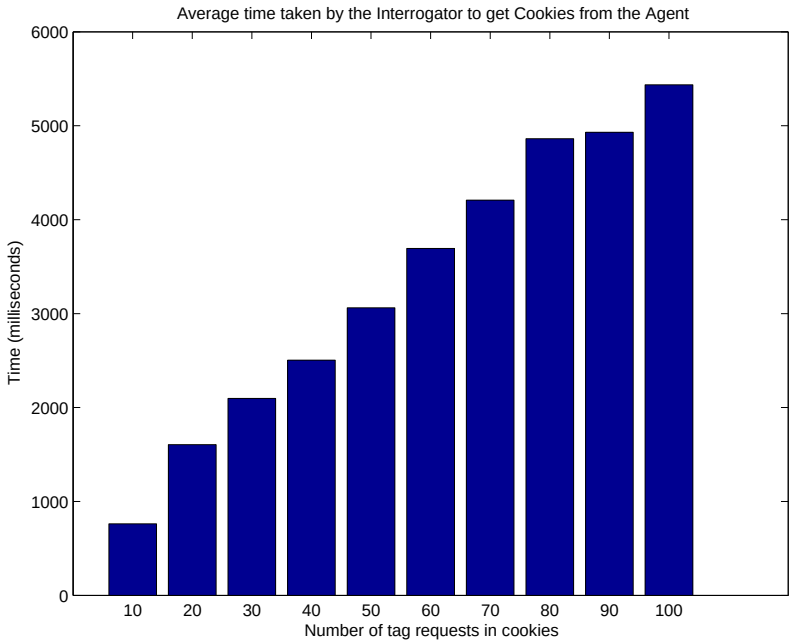

FIG. 7.11. Time taken by Interrogator to get Cookie from Authenticating Agent

This step is described in Fig 7.4. As described in Section 6.3, the Interrogator will communicate with the Agent to get a cookie for the tags it wants to get details about. We assume that tags can be queried by their meta data. The meta data is the description chosen by the Dispatcher to describe the item being sent. Fig 7.11 describes the time taken by an Interrogator to communicate with the Agent for receiving a cookie corresponding to the tags of interest. As observed in the graph, the amount of time increases with the number of requests because more data is being transferred per request as shown in Fig 7.12 as well as includes network delays. We that the cookie contains handles to the requested tags with the sentinel hence has smaller data size compared to the request. The minor variations in the increase of data transferred/time taken with increasing number of tags can be explained as follows: The amount of data

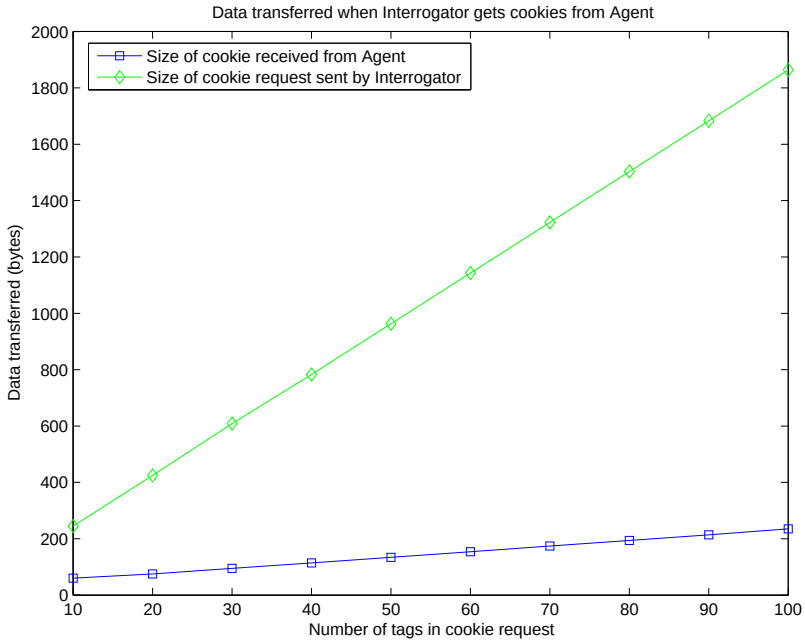

FIG. 7.12. Data transfer involved in an Interrogator getting a Cookie from the Authenticating Agent

being sent by the Interrogator changes with the length of meta data for each query. Similarly, if a non-existent meta data is being queried for, the Interrogator would issue a dummy handle in the cookie, in place of an actual handle to the tag in the cookie.

2. Interrogator gets decoding keys from the Sentinel

   This step is described in Fig 7.5. The Interrogator would then need to present the received cookie to the Sentinel. The Sentinel would validate the cookie and issue it the decoding keys. The Sentinel issues the encoded identifier in addition to the decoding key. If the Sentinel would not issue the encoded identifier, the Interrogator would have to apply the decoding key to all the encoded identifiers that it has read, and then run a function on them to verify which of the identifiers it has successfully decoded. As observed in Fig 7.15, the time taken by the Sentinel to respond with

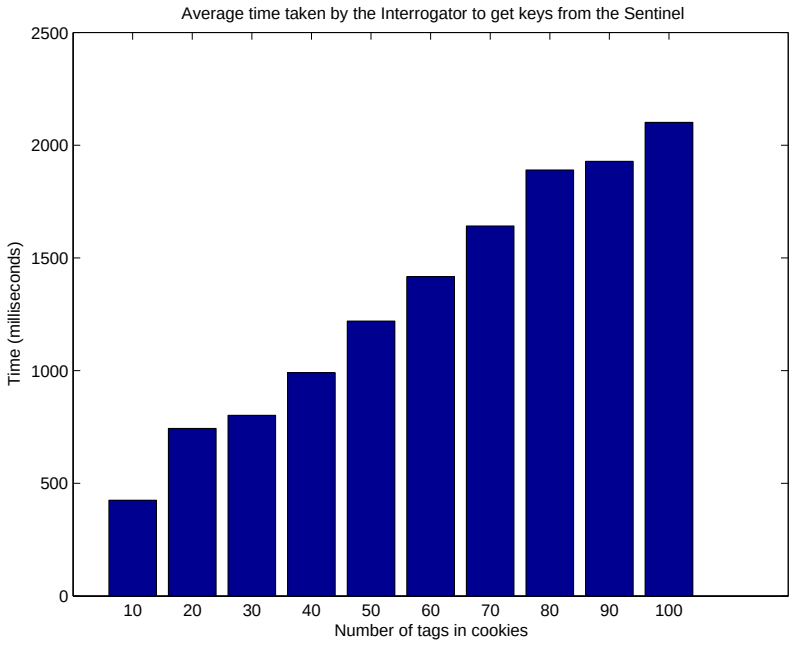

FIG. 7.13. Time taken by Sentinel to issue decoding keys to Interrogator

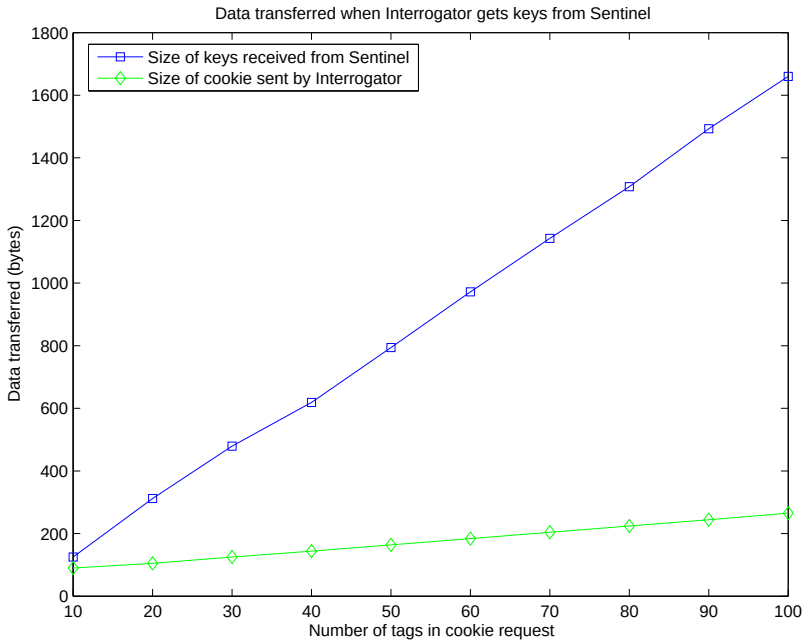

FIG. 7.14. Data transfer involved in an Interrogator getting decoding keys from the Sentinel

the tag passwords increases with the number of tag requests that are encoded in the cookie. This is because the data transferred increases with the number of tag requests, as shown in Fig 7.14 as well as includes the network delays. We note that more data is transferred from the Sentinel back to the Interrogator as in response to each valid tag handle in the cookie, the sentinel issues the encoded identifier and the decoding key as well. The small variations in the increase in data transfer/time taken may be attributed to the following: If a dummy handle is present in the cookie, the Sentinel will issue a blank encoded identifier and corresponding decoding keys.

3. Interrogator gets the tag details from the Authenticating Agent

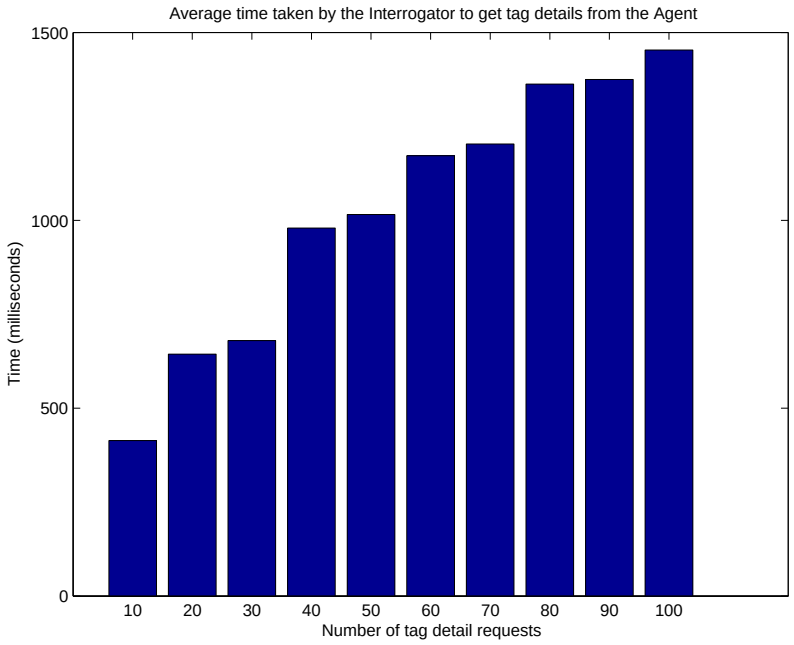

FIG. 7.15. Time taken by Authenticating Agent to issue tag Details to Interrogator

This step is described in Fig 7.6. After receiving the decoding keys, the Interrogator decodes the tags with the corresponding keys, and retrieves the original identifier of the tags. To get more details of the tag, the Interrogator needs to present the original

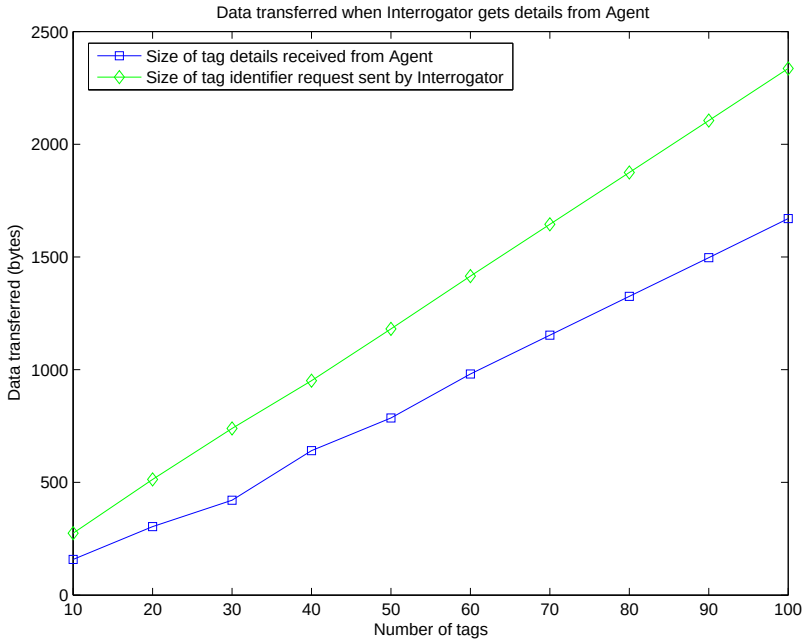

FIG. 7.16. Data transfer involved in an Interrogator getting tag details from the Authenticating Agent

identifier to the Authenticating Agent. The time required for this interaction is shown in Fig 7.15. The time required increases with the number of tags and increasing data transfer as shown in Fig 7.16 along with network delays. The input to the Agent is the decoded identifier of the tag, whereas the response from the agent is the detail of each valid tag. The variations in increase are due to the level of detail permitted to the Interrogator. Depending on the access levels of the Interrogator, different details of information will be sent to the Interrogator. Hence the amount of data transferred for each tag changes. Similarly, if a non-existent tag identifier is queried, the Agent replies with an “Access denied.” response.

We describe the sum of the timings and data transfer that take place in the three steps taken by the Interrogator to get the tag details as follows:

The overall/total time taken by the Interrogator to learn the tag details is shown in

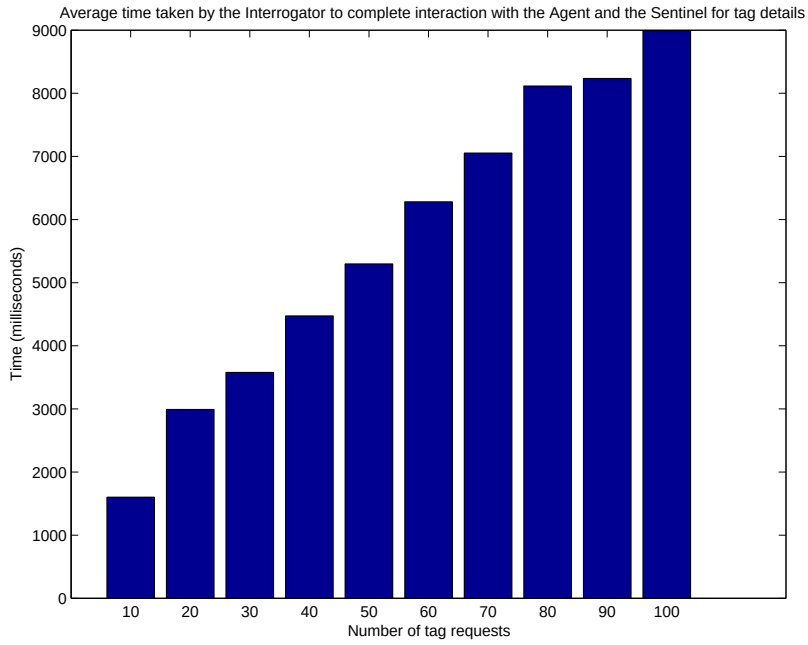

FIG. 7.17. Total Time taken by Interrogator to get access to tag details

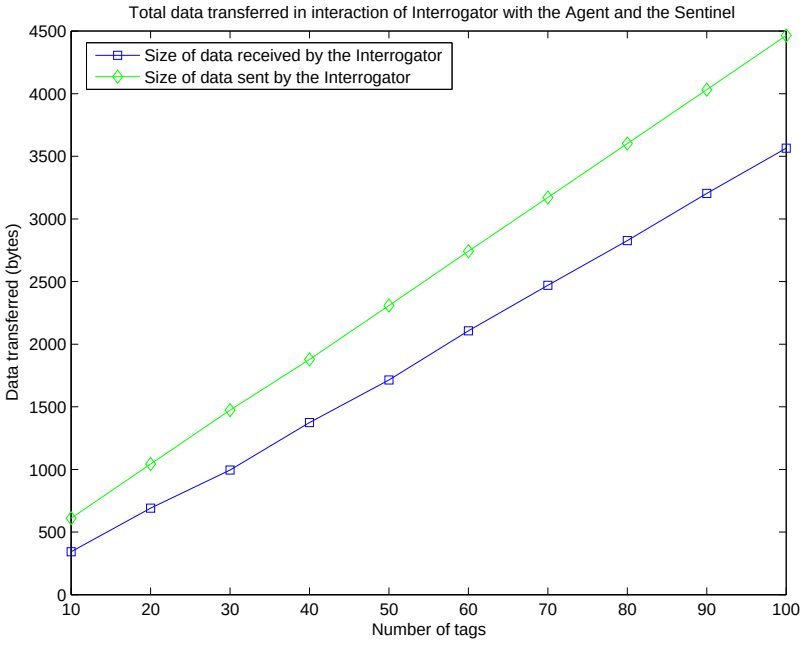

FIG. 7.18. Total data transfer in interactions of the Interrogator with the Authenticating Agent and the Sentinel

Fig 7.17. This is the sum of the time required by Interrogator to interact with the Sentinel and Agent as explained above. We observe that the time needed increases with the number of tags for which information is requested, and is affected by the factors that affect the respective communication with the Agent and the Sentinel. The total data transfer for the above steps is shown in Fig 7.18, and is affected by the factors which affect the individual steps in the data transfer.

Chapter 8

SECURITY ANALYSIS

8.1 Scenarios

Before we analyze our mechanism from the security perspective, we identify the different scenarios in which security is important.

1. Loading of goods onto a truck.
2. Unloading goods from a truck.
3. Hand off of goods from one Sentinel to another
4. Individual item being added to the truck
5. Individual item being offloaded/delivered from truck

For the loading/unloading of goods on/off the transit vehicle, the Sentinel would need a cue (from perhaps the vehicle driver) that an authorized pickup (or shipment) of items(s) has been made. When goods are first loaded onto the truck, the goods would be having their original identifiers. The sentinel would need to contact the Agent to inform the Agent about the the observed goods. The Agent would then issue the passwords of the respective goods to the Sentinel and note against each good the Sentinel identification. The Agent would also issue a set of encoding keys to the Sentinel. Similarly, at the time of offloading the

goods, the Sentinel would communicate to the Agent the intent of restoring tag identifiers. The Agent would update its database, as well as if needed, instruct the Sentinel to change passwords for the goods.

Similarly, when the goods are moved from one transit truck to another (or the truck passes over from one Sentinel area to another), the Sentinel would notify the Authenticating Agent of the change, so that the Authenticating Agent could program the Sentinel on board the next transit truck for the goods. Similarly, in the event of the Sentinel going offline, the Authenticating Agent can recover the identity of the goods. The Authenticating Agent has the set of keys from which it can recover the goods, it can hand off to a new Sentinel the passwords and possible keys of the goods; this Sentinel can try all possible combinations of the keys to recover the valid tag identifiers.

The case of individual goods being loaded/unloaded from the truck is similar to loading/unloading all goods, except that the Sentinel would need some kind of guarantee that the loading/unloading action is legitimate and the the tag identifiers have not been spoofed to trigger a load/unload event.

8.2 Resilience to Attacks

We consider the impact of different kinds of attacks on the system.

8.2.1 Computational Attacks

In the computational aspect, an Interrogator having access to multiple keys also does not pose a threat. This is because each key matches just one tag and interrogator cannot get any useful information about any other tag. Further, keys are temporal and would not be re-used, thus thwarting reverse-engineering attempts on a set of keys.

8.2.2 Misrepresenting identity of Agent, Sentinel or Interrogator

We assume that each of the Agent, Sentinel and Interrogator have digital certificates which are issued by a trusted certificate authority, so each of the entities in the interaction could verify the certificate. We assume that the certificates cannot be stolen or tampered.

8.2.3 Attacks on tags

In terms of RFID entities, these attacks include attempts by adversaries to write to tags, copying data off tags, possibility tampering tag data and spoofing tags. A swapping attack on the data identified in [7] would be one in which an Interrogator swaps the EPC identifiers on the RFID tags. We rule out this attack since we assume that only the Sentinel can write the data back to the tags (assuming the existence of password protected write for the tag). Our scheme similarly does not suffer from the problem of re-encryption (in which the original identifier may get corrupted), since the decoding keys are used to retrieve the original identifier before re-encrypting it. Copying of tag data (and possibly Sentinel digital signature) off the tag does not constitute an attack. This is because even if a new tag outside the truck is programmed by the adversary, the truck being physically secure, new goods / tags cannot be placed inside the truck. Although a rogue reader cannot write a tag without knowing the password, if a rogue reader hacks a tag to retrieve the password as described by Shamir et al. in [20], a rogue reader may overwrite or corrupt the tag data. This is a Denial-of-Service attack on in which adversary actually learns nothing about the tags, but prevents the Sentinel from following its reprogram protocol.

8.2.4 Attack on truck or physical tampering of tags

Our security enhanced variant of the scheme using active tags would be able to counter this threat. Active tags can have sensors onboard, and since they are continuously powered, they could endure that they are not being physically tampered, or the system is not under

attack, e.g. using a heat sensor to measure ambient temperature. The active tag could alert the Sentinel or the Agent about the attack.

8.3 Shortcomings

We assume that attacks on physical security of the transit vehicle or the contained RFID tags are not possible, and further our system cannot defend against an attack involving use of “Radio finger-prints” of RFIDs as mentioned in [18], or attacks that disrupt RFID communication. We note that the Sentinel has keys for all the tags it is protecting. In case the Sentinel is compromised, yielding all keys presently with the Sentinel, an attacker could potentially decode and retrieve the original identifier for the tags. We assume that the Sentinel would have some kind of signaling mechanism to alert the Authenticating Agent. Similarly in case of attacks on tags including side-channel attacks, like [20] which reveals the tag password, we could require that a write to the tag needs special hardware available only at the Sentinel, an active tag could alleviate this problem by making a distinction between a Sentinel and other devices. Another extreme case in which an adversary continually copies (and updates on every reprogram cycle) all tag identifiers and reprograms fake goods on another truck, we would need a method by which we can tell the two trucks apart, perhaps using the active tags.

Chapter 9

FUTURE WORK

In the future, we would try to make the system more intelligent by making the Sentinel context-aware, and equipping it with a GPS system to monitor the movement of the truck. Similarly, using capabilities of active tags like temperature sensor, displacement sensor, it would be possible to determine if the items in the transit have been moved, or if there are any situations in the truck which require immediate attention. The Authenticating Agent could also have an ontology based system to help if identify the various interrogators to learn their access levels. From a data mining perspective, the Authenticating Agent could maintain a log of all the interactions of the Sentinel en route and use it to find trends in the entities encountered, and perhaps suggest an alternative route for the transit. We could fine tune the parameters of our system depending on the products which are being shipped. For example, if medicines are being shipped, one can use passive tags with frequency ranges which do not get affected by fluids. In case of high cost items like sensitive documents, or a military consignment, we could deploy active tags with the highest levels of security to ensure visibility of the items and the safety of the vehicle at all times.

Chapter 10

CONCLUSION

With RFID rapidly becoming a pervasive technology, the security and privacy considerations of RFID tags are paramount. With commodities as varied as bank notes, airport luggages, clothing items, the privacy aspects of each system need to be individually addressed. For example, it may not be advisable to allow writes to RFID tags like those on E-Z Pass [21], or bank notes.

The scheme we have presented represents a complete solution to protecting the identity of RFID tags for goods in transit. We can use encryption and digital signatures in our scheme despite the use of passive read/write tags, which are resource constrained. We use simple schemes to security in RFIDs. This scheme can be deployed without any changes to the existing EPC Class 1 architecture and is cost-effective. Further, it can be integrated with the EPC Network. Beyond the fixed known locations in warehouses, once goods are in transit, the Sentinel can replace the tag identifiers with its own and synchronize its changes with (a slightly modified version) of the ONS.

Our scheme thus preserves the theme of RFID as a technology to provide quick access to information, at the same time protecting it from being misused. RFID can blend in with ubiquitous computing with its form factor, ubiquitous location and varied fields of application. In combination with a common ontology, a means of interpreting data from different sources including RFIDs can be developed.

REFERENCES

[1] "Is RFID the Supply Chain Holy Grail?." url-http://logistics.about.com/od/rfid/a/aa122404.htm.

[2] "EPCglobal Inc. Home Page." urlhttp://www.epcglobalinc.org/.

[3] EPC Global Inc., "The EPCglobal Network:Overview of Design, Benefits, & Security." urlwww.epcglobalinc.org/news/EPCglobal Network Final 09 24 004 Final.pdf.

[4] "Nokia 5140 RFID Reader." urlhttp://www.mobilemag.com/content/100/104/C2607.

[5] Marks and Spencers, "Background to Marks & Spencer's business trial of RFID in its clothing supply chain." url-http://www2.marksandspencer.com/thecompany/mediacentre/ pressreleases/2005/com2005-02-18-00.shtml.

[6] "UPS Pressroom: Fact Sheets." urlhttp://www.pressroom.ups.com/mediakits/factsheet /0,2305,1202,00.html.

[7] A. Juels, "RFID Security and Privacy: A Research Survey," in *IEEE Journal on Selected Aread In Communications, Vol. 24, No. 2*, pp. 381–394, 2006.

[8] RSA Security Inc., "Technical Characteristics of RFID,"

[9] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong Authentication for RFID Systems Using the AES Algorithm," in *Proc. Cryptographic Hardware and Embedded Systems 2004 - 6th Int. Workshop*, pp. 201–212, 2004.

[10] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems," in *Security in Pervasive*

Computing: 1st Int. Conf., Boppard, Germany, March 12-14, 2003., pp. 201–212, 2004.

[11] EPC Global Inc., "EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz." url"http://www.epcglobalinc.org/standards-technology/EPCglobalClass-1Generation-2UHFRFIDProtocolV109.pdf.

[12] R. Want, "Introduction to RFID Technology," in *Pervasive Computing, Vol. 5, Issue 1*, pp. 25–33, 2006.

[13] "RSA Security - Technical Characteristics of RFID." url-http://www.rsasecurity.com/rsalabs/node.asp?id=2121.

[14] "Supply-Chain Technology, Track(ing) into the Future , The Impending RFID-Based Inventory Revolution," in *BearSterns Report*, pp. 619 – 623, 2003.

[15] " EPC Generation 1 Tag Data Standards Version 10 1.1 Rev.1.27 ." url-http://www.epcglobalinc.org/standards_technology/ EPC_TDS_1 1_Rev_1 27_Ratification_final 1 2006.pdf.

[16] "4. Information technology AIDC techniques - RFID for item management Part 3: - Parameters for air interface communications at 13.56 MHz," in *ISO/IEC JTC 1/SC 31/WG 4*, 2004.

[17] A. Juels and R. Pappu, "Euros Squealing Euros: Privacy Protection in RFID-Enabled Banknotes," in *LNCS 2742*, pp. 103–121, 2003.

[18] G. Avoine and P. Oechsli, "RFID Traceability: A Multilayer Problem," in *IFCA, Financial Cryptography, Pre-Proceedings version*, 2005.

[19] Z. Kfir and A. Wool, "Picking virtual pockets using relay attacks on contactless smartcard systems." urlhttp://eprint.iacr.org/2005/052.

[20] "Epc tags subject to phone attacks." urlhttp://www.rfidjournal.com/article/articleview/2167/1/1/.

[21] "E-Z Pass." urlhttp://www.ezpass.com/.

[22] Stephen August Weis, "Security and Privacy in Radio-Frequency Identification Devices," Master's thesis, 2003.

[23] D. Molnar and D. Wagner, "Privacy and security in library RFID: Issues, practices,and architectures," in *Conference on Computer and Communications Security, ACM CCS* , pp. 210 – 219, 2004.

[24] A. Juels and R. Pappu, "Squealing Euros: Privacy protection in RFID-enabled banknotes," in *Proc. Financial Cryptography, R.Wright, Ed. New York: Springer-Verlag, vol. 2742, Lecture Notes in Computer Science*, pp. 103–121, 2003.

[25] P. Golle, M. Jakobsson, A. Juels, , and P. Syverson, "Universal re-encryption for mixnets," in *Proc. RSA Conf. -Cryptographers Track (CTRSA), T. Okamoto, Ed., vol. 2964, Lecture Notes in Computer Science*, pp. 163–178, 2004.

[26] A. Juels, R. Rivest, and M. Szydlo, "The blocker tag: Selective blocking of RFID tags for consumer privacy," in *Conference on Computer and Communications Security, ACM CCS*, pp. 103 – 111, 2003.

[27] A. Juels, "Minimalist Cryptography for RFID Tags," in *C. Blundo. Ed., Security of Communication Networks (SCN)*, 2004.

[28] L. Bolotnyy and G. Robins, "Randomized pseudo-random function tree walking algorithm for secure radio-frequency identification,"

[29] "Veriwise asset intelligence." urlhttp://www.geveriwise.com.

[30] I. Satoh, “Linking Physical Worlds to LogicalWorlds with Mobile Agents,” in *Proceedings of the 2004 IEEE International Conference on Mobile Data Management*, pp. 332–343, 2004.

[31] “Active and Passive RFID: Two Distinct, But Complementary, Technologies for Real-Time Supply Chain Visibility.” urlhttp://www.autoid.org/2002_Documents /sc31_wg4/docs_501 520/520_18000 7_WhitePaper.pdf.

[32] “AR400 RFID Reader from Symbol.” urlhttp://www.symbol.com/products/ rfid/data_sheet_ar400.html.

[33] “Alien Technology - RFID Tags.” urlhttp://www.alientechnology.com/products/rfid_tags.php.

[34] K. M. Ramakrishnan and D. D. Deavours, “Performance Benchmarks for Passive UHF RFID Tags.” urlhttp://www.ittc.ku.edu/ deavours/pubs/mmb06.pdf.

[35] VeriSign, Inc., “The EPCglobal Network: Enhancing the Supply Chain.” urlhttp://www.verisign.com/static/002109.pdf.