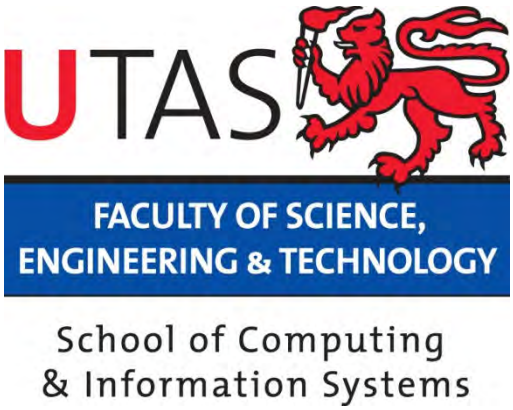

A Whole of System Approach to Analysis of Security in RFID Systems Using an Integrated Layered and Partitioned Reference Model

by

Luke Thomas Mirowski, BComp. (Hons)

Submitted in fulfilment of the

requirements for the Degree of

Doctor of Philosophy

University of Tasmania

(June, 2011)

I STATEMENT OF ORIGINALITY

This thesis contains no material which has been accepted for a degree or diploma by the University or any other institution, except by way of background information and duly acknowledged in the thesis, and to the best of my knowledge and belief no material previously published or written by another person except where due acknowledgement is made in the text of the thesis, nor does the thesis contain any material that infringes copyright.

Date:

II STATEMENT OF AUTHORITY OF ACCESS

This thesis could be made available for loan and limited copying in accordance with the *Copyright Act 1968*.

Date:

III STATEMENT OF CO-AUTHORSHIP

The publications of the work undertaken in the course of this research are the following:

Mirowski, L., J. Hartnett and R. Williams (2009). "An RFID Attacker Behavior Taxonomy." IEEE Pervasive Computing **8**(4), pp. 79-84.

- Mr. Luke Mirowski (80%) is the primary author.
- Mrs. Jacqueline Hartnett (10%) and Dr. Raymond Williams (10%) of the School of Computing and Information Systems, University of Tasmania, provided general guidance and editing advice as supervisors.

Mirowski, L., J. Hartnett and R. Williams (2009). How RFID Attacks are Expressed in Output Data. Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN 2009), Kaohsiung, Taiwan, pp. 794-799.

- Mr. Luke Mirowski (80%) is the primary author.
- Mrs. Jacqueline Hartnett (10%) and Dr. Raymond Williams (10%) of the School of Computing and Information Systems, University of Tasmania, provided general guidance and editing advice as supervisors.

Mirowski, L., J. Hartnett and R. Williams (2009). Tyrell: an RFID Simulation Platform. Proceedings of the Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2009), Melbourne, Australia, pp. 325-330.

- Mr. Luke Mirowski (80%) is the primary author.
- Mrs. Jacqueline Hartnett (10%) and Dr. Raymond Williams (10%) of the School of Computing and Information Systems, University of Tasmania, provided general guidance and editing advice as supervisor.

Mirowski, L., J. Hartnett, R. Williams and T. Gray (2008). A RFID Proximity Card Data Set: Technical Report, School of Computing and Information Systems, University of Tasmania.

- Mr. Luke Mirowski (70%) is the primary author.
- Mrs. Jacqueline Hartnett (10%), Dr. Raymond Williams (10%), and Mr. Tony Gray (10%) of the School of Computing and Information Systems, University of Tasmania, provided general guidance and editing advice as supervisors and consultant.

We the undersigned agree with the above stated proportion of work undertaken for each of the above published manuscripts contributing to this thesis.

Signed: ……..

Date: ……..

Mrs. Jacqueline Hartnett

Supervisor

School of Computing and Information Systems

University of Tasmania

Signed: ……..

Date: ……..

Dr. Raymond Williams

Supervisor

School of Computing and Information Systems

University of Tasmania

Signed: ……..

Date: ……..

Mr. Anthony Gray

Consultant

School of Computing and Information Systems

University of Tasmania

IV ABSTRACT

This thesis proposes the use of a 'whole of system' approach to the analysis of security in Radio Frequency Identification (RFID) systems and introduces a reference model for this purpose. It illustrates the advantages of this approach in the context of detecting clone tags within RFID systems, including the specific example of a pharmaceutical supply chain. It compares the results from using the proposed model with those from previous work that adopted a more localised approach (Rotter 2008; Mitrokotsa et al. 2010). In order to enable the 'whole of system' approach, a domain model for RFID systems is introduced and a simulator based on this is implemented. Interesting insights arising from simulator results are confirmed through laboratory experiments.

The reference model proposed consists of the three horizontal layers suggested by previous authors: real world, RFID and strategic (Mitrokotsa et al. 2008, 2009), but adds vertical security partitions for such things as the problem context. This provides a structure that allows existing analysis methods from any appropriate source to be applied systematically, such that their results are integrated across the whole system. It is shown that this provides for the analysis of not only the security requirements of the whole system but also, where in the system it is practicable to place measures that achieve these requirements.

The domain model introduced comprises a logical view of RFID components and a data view of the associations and features that characterise the component interactions. The model's controlled vocabulary allows the domain constructs in RFID systems to be identified and described. A simulator, which has been validated for preliminary 'whole of system' analysis and is based on this domain model, allows experimentation with systems via an application programming interface (API).

Work suggested by the reference model is reported as simulation results, and confirmed by laboratory experimentation using Class-One Generation-Two RFID equipment. Whereas Juels (2005) showed that tags of this standard can be reprogrammed so that they can be authenticated by a reader, the results in this thesis illustrate how readers can be reprogrammed to expose clone tags, thereby contributing an additional security solution.

This thesis addresses an analysis gap in the RFID security field by introducing a 'whole of system' approach made possible by the proposed reference model. The results illustrate that the effectiveness of security in RFID systems can be improved by employing a range of individual analysis methods integrated into this model.

V ACKNOWLEDGEMENTS

I am grateful for the help provided by a number of people during the completion of this thesis.

I would like to acknowledge my PhD supervisors, Jacqueline Hartnett and Mike Cameron-Jones. Jacky's knowledge, insights, and enthusiasm over the years were instrumental to the completion of this thesis. Mike's rigorous form of analysis and guidance ensured this thesis reached its full potential.

I would also like to acknowledge my former PhD supervisors, Raymond Williams and Christopher Lueg. Ray's approach to problem solving resulted in a number of interesting discoveries throughout this thesis. Christopher's early influences ensured this thesis took on a broader focus.

I am grateful for the assistance provided by Mark Anderson and Ian Whitehouse, whose insights into the RFID field, as well as provision of data and equipment, were much appreciated.

Without the support of my fellow PhD's from the School of Computing and Information Systems, this work would have been far more challenging to complete. I am especially grateful to Joel Scanlan, Ivan Bindoff, Tristan Ling, and Matthew Armsby for their time and effort in providing research advice.

Countless others have influenced this work in many ways. To those people I am truly appreciative.

Finally, this thesis would not have been possible without the love and support of my family to which I am forever indebted.

VI TABLE OF CONTENTS

VII LIST OF TABLES

VIII LIST OF FIGURES

Chapter 1

Introduction

1.1 INTRODUCTION

Radio Frequency Identification (RFID) has been used to identify objects for over 50 years (Garfinkel and Holtzman 2005). During World War Two, the Identification Friend or Foe (IFF) system allowed for Allied aircraft to be distinguished from the enemy on the basis of coded radio signals sent from transponders to a base station. Years later, advances in electronics led to commercial adoption (Landt 2005). Electronic Article Surveillance (EAS), developed by Sensormatic, Checkpoint, and Knogo in the 1960's was an early commercial RFID system, used to counter the theft of merchandise. Rapid growth in commercial use of RFID occurred in the late 1980's when electronic toll collection was introduced in Europe and then the United States. Since then RFID has undergone significant developments.

Nowadays, the basic premise of RFID is that objects are marked with tags which emit serial numbers obtainable by readers using radio signals (Weinstein 2005). When compared with barcode technology, RFID identifies objects without requiring line of sight. Once a tag's serial number has been obtained, a reader retrieves information about the serial number from a database, and acts upon it accordingly. Tags fall into two general categories: active or passive. Active tags contain their own battery power making them physically large and expensive. Conversely, passive tags obtain their power from the signal of a RFID reader, and are therefore, usually small and low cost. Consequently, passive tags are expected to be more widely adopted then active tags and will dominate the widespread adoption of RFID into the future. (Weinstein 2005)

Electronic Product Code (EPC) technology, developed by the Auto-ID Center, established at MIT in 1999 and now managed by EPCglobal, is leading the widespread adoption of RFID technology into various operations (Garfinkel and Rosenberg 2005). Electronic Product Code (EPC) technology was introduced as an extensible range of tag standards, of which, the Class-One standard represented a cost effective and widely accepted design. It was mandated for use as a Supply Chain Management (SCM) technology by Wal-Mart and the United States Department of Defence (DoD), and its later standard, Class-One Generation-Two was ratified as an international standard, ISO 18000-6c, a few years later (Roberti 2004). This, along with rapid growth in item level tagging in the retail apparel

industry, has led to a surge in EPC Generation-Two integrated circuit (IC) sales volume, which is expected to exceed one billion units in 2010 (Swedberg 2010b). This has led Bill Colleran, chief executive officer of Impinj, among the first companies to introduce products based on the EPCglobal UHF Generation-Two standard (O'Connor and Roberti 2005), to speculate that the widespread use of these tags in the clothing industry is only a few years away, leading the way for its widespread adoption in other industries.

A recent study by ODIN Technologies, reported by news sources (Swedberg 2010a), has indicated that EPC Class-One Generation-Two tags remain at a price of around 15 cents apiece for quantities of 10,000 and 11 cents each when ordered in quantities of one million or more, however; years ago Sarma (2001) speculated that tags would need to cost as little as five cents each before they could begin to become widely adopted. He speculated that to achieve this cost, such tags would be nothing more than integrated circuits equipped with a radio antenna and a serial number. If we follow the argument of Sarma (2001), it may mean that tags undergo some reductions in functionality to achieve a cost of five cents. Therefore, tags which may become widespread would be relatively simple devices, with a minimum level of functionality for such things as security.

Widespread adoption of RFID will bring many benefits to industry, but at the same time, the need for security can be seen in the following application examples, which highlight a variety of RFID uses, but illustrate the varying security requirements of different systems:

- Since the United States Federal Drug Administration (FDA) recommended the use of RFID (FDA 2004), news sources have reported trials of ISO-15693 tags by GlaxoSmithKline on bottles of Trizivir (O'Connor 2006), (a Human Immunodeficiency Virus (HIV) medicine), for preventing the introduction of counterfeit drugs. However, Hancke (2005) demonstrated ISO-15693 tags are vulnerable to relay attacks, implying the possibility that even with RFID, counterfeit Trizivir could still be introduced into the system.

- The MiFare Classic inlay is a smart-card which uses RFID technology in over 200 million tags, in systems like the London Oyster Card system, and the

Dutch OV-Chipkaart system. The MiFare Classic's encryption algorithm and authentication protocol can be reverse engineered, revealing ways an attacker can read a card, clone a card, or restore a card to a previous state to commit payment fraud (Garcia et al. 2008).

- Finally, the Texas Instruments Digital Signal Transponder (TI-DST), used in over 150 million vehicle immobiliser keys and the ExxonMobil Speedpass system, has been cloned by a research team from John Hopkins University. A cloned tag was used to purchase petrol in the Speedpass system, and to spoof the immobiliser authentication system of a 2005-model Ford Escape sport utility vehicle (SUV). (Bono et al. 2005)

Fundamentally, the link between the object, the tag, and the reader is one based on surrogacy. As radio frequency communication is non-contact and non-line-of-sight, it is more difficult for the owner of a tag to validate this relationship (Sarma et al. 2003). Trust in this system could be achieved if cryptography could fit into a label's functionality without dramatically increasing the cost of the label (Ranasinghe et al. 2004). However this seems unlikely for the tags which are envisaged to become widespread, as the number of gates available in these tags is 4000 gates or less. Unfortunately, for private key cryptosystems such as the Advanced Encryption Standard (AES), a commercial implementation of AES typically requires 20,000 to 30,000 gates (Sarma et al. 2003). While other onboard options for cryptography have been suggested for authentication of tags or readers, such as modified hashing algorithms, these do not meet the cost requirements of these tags.

The examples above expose the need for structured analysis of security issues in the widespread adoption of RFID technology. This has led to the research topic explored in this thesis: the use of a 'whole of system' approach to the analysis of security in RFID systems.

Methods for analysis of security in RFID systems currently exist; however these take a relatively localised view of security. Rotter (2008) proposed a privacy and security risk assessment framework which was used to assess domain risks using three criteria: the system's deployment range; the link between the RFID tag and identity-related data; and the domain's security demands. Conversely, Mitrokotsa et al.

(2008, 2009) structured threats into system layers, enumerating the threats as well as offering potential defences for each layer. The model they introduced for this purpose discriminates attacks by layers: physical, network-transport, application, and strategic. Finally, work by Mitrokotsa et al. (2010) extended the concept of assessing security using layers, introducing the concept of security principles at each system layer in addition to attribute columns for such things as cost and potential damage. When considering these examples, it seems likely that an improvement to be made is the capacity for capturing sufficient system information which would enable the derivation of security requirements which consider the 'whole system'.

Consequently, this thesis addresses an analysis gap in the RFID security field by introducing a 'whole of system' approach to analysis, made possible by way of a reference model. This model consists of the three horizontal layers suggested by previous authors: real world, RFID and strategic (Mitrokotsa et al. 2008, 2009), but adds vertical security partitions for such things as the problem context. This provides a structure that allows existing analysis methods from any appropriate source to be applied systematically, such that their results are integrated across the 'whole system'.

Throughout this thesis a central theme will be challenging the established analysis approaches which currently exist in the field. Using the model together with various analysis techniques (for example: domain modelling, entity-relationship modelling, and attack trees) illustrates the advantages of the proposed approach in the context of detecting clones within RFID systems, including the specific example of a pharmaceutical supply chain. Results from using the proposed model are compared with those from previous work that adopted a more localised approach. Overall, the results illustrate that the effectiveness of security in RFID systems can be improved by employing a range of individual analysis methods integrated into this model, which is not facilitated by existing approaches.

1.2 MOTIVATION

The work reported in this thesis has emerged out of the need to develop practicable security to prevent or detect tag cloning. Tag cloning, one problem explored throughout this thesis, allows an attacker to duplicate a tag's identification data. A

clone tag can be assigned to an unauthorised entity, thereby allowing the tag to derive an authorised entity's privileges in a system (Juels 2005). For systems which rely on tags for authentication, like the system reportedly in use by GlaxoSmithKline on bottles of Trizivir (O'Connor 2006), the ability to clone tags, implies counterfeit Trizivir could enter the system and remain undetected even though an RFID system is in use.

Early research on developing an intrusion detection system capable of detecting clone tags illustrated the advantages of placing security in the middleware - where security is usually more cost effective than on the tag. The system used statistical profiling of tag data to identify instances which represented when a tag had change ownership between entities (Mirowski 2006). Considered as 'state of the art' by Lehtonen et al. (2007b), it was proposed to be used in places where cryptography on tags was not practical. The derivation of practicable solutions, however, demands knowledge of a system's security requirements.

1.3 KEY CONTRIBUTIONS

The key contributions made by this thesis are as follows:

- Illustrating the advantages of taking a 'whole of system' approach to analysis of security in RFID systems by employing a range of individual analysis methods which are integrated into a reference model. (Chapter 6 - Chapter 10).

- Introducing a model for security analysis which provides a structure that allows existing analysis methods from any appropriate source to be applied systematically, such that their results are integrated across the whole system. (Chapter 5).

- Presenting a domain model which comprises a logical view of RFID components, in addition to a data view of the associations and features that characterise the component interactions. The domain model's controlled vocabulary allows the domain constructs in RFID systems to be identified and described. (Chapter 6).

- Enumerating RFID threats over system layers which illustrates that a systematic approach to threat analysis can assist in identifying attacks, and where it is more effective to address threats in a system. (Chapter 7).

- Contributing a simulator, validated for preliminary 'whole of system' analysis, which has been based on a domain model. It also implements the model's controlled vocabulary via its application programming interface (API) allowing for repeatable systems modelling. (Chapter 8).

- Suggesting that a reference model based approach gives impetus to interesting simulation results, and confirming these through laboratory experimentation using Class-One Generation-Two RFID equipment. Laboratory work also illustrates how readers can be reprogrammed to expose clone tags, thereby contributing an additional security solution (Chapter 9).

- Illustrating the advantages of a 'whole of system' approach in the context of the specific example of a pharmaceutical supply chain. When results from using the proposed model are compared with those from previous work that adopted a more localised approach, the effectiveness of the approach is illustrated. (Chapter 10).

1.4 THESIS OUTLINE

What follows is the structure of this thesis.

- Chapter 2 reviews previous work on Radio Frequency Identification (RFID) security in the context of cloning attacks and system constraints. Considering that cloning has system-wide consequences and various methods for enactment, and because of system constraints, it seems likely that achieving practicable security for cloning requires systematic analysis of security requirements.

- Chapter 3 reviews previous work on approaches to security analysis in RFID beginning with a review of standard-system models. The properties common to systems are discussed, and these are used to consider the closeness of existing security models to system concepts. In considering these security

models, it seems that existing work has focussed on a relatively localised view of security. Consequently, it seems likely that these examples have a limited capacity to distil more relevant information that contributes to more explicit security requirements in actual systems.

- Chapter 4 reviews a range of individual analysis methods suitable for proposing an alternative model for the proposed 'whole of system' analysis approach. These methods come from a variety of backgrounds, but are focussed on systematic analysis. The reference model paradigm suggests a method for model derivation, whereas other methods are envisaged as useful, when integrated into a reference model, for the purpose of specific modelling tasks in RFID security analysis.

- Chapter 5 introduces an alternative model for security analysis in RFID systems, based on the reference model approach. When considering the examples of previous work, this model is distinguished on the basis of integrated layer and partition properties, and is therefore entitled, *An Integrated Layered and Partitioned Reference Model*. Its qualities are discussed along with its intended use to enable a 'whole of system' approach to analysis using existing analysis methods.

- Chapter 6 illustrates how analysis using the standard operating partition of the reference model facilitates the enumeration of system elements into a domain model. Various domain analysis methods are integrated to define a logical view of components and these are modelled using the Unified Modelling Language (UML). Following this, a data view of the system associations is derived using Entity-Relationship Diagramming (ERD) and from this, new features are constructed. What emerges is a domain model and controlled vocabulary which allow for domain constructs to be identified and described. This forms the basis for security analysis in a systems context.

- Chapter 7 illustrates how analysis using the problem partition of the reference model, enables the systematisation of attacks. The attack tree method is applied to two system types, and attacks are organised as a hierarchy which

broadly matches RFID system layers. The chapter makes practical use of the taxonomy to identify good locations for solutions in the example problem of the Trizivir pharmaceutical system. As the taxonomy is relatively generic, its broader potential for use is discussed.

- Chapter 8 introduces a software simulator based on the domain model, which has been validated for 'whole of system' analysis. It is applied to the analysis of solutions in the reference model's solution partition. The simulator is used to explore the example solution of tag and reader associations for attack exposure. The benefits of taking a systems approach to solution analysis are discussed using this example, while impetus for exploring some of the results further with EPC Class-One Generation-Two equipment is identified.

- Chapter 9 demonstrates, through experimentation with EPC Class-One Generation-Two equipment, how results obtained following a 'whole of system' analysis, first explored conceptually through the model, and then in simulation, can be realised in a system. The results of simple experimentation, confirm the results of simulation, while illustrating the possibility, in systems which use the Slotted Random Anti-Collision (SRAC) protocol, of exposing clone tags through the reprogramming of an RFID reader. The findings are discussed in relation to the need for appropriate systems analysis prior to solution deployment.

- Chapter 10 validates the 'whole of system' approach in the context of the specific example of a pharmaceutical supply chain. The validation takes place in each partition, while the results are integrated across the whole model. It compares the results from using the proposed model with those from previous work that adopted a more localised approach. These results suggest that a 'whole of system' approach made possible by the proposed reference model leads to more effective security requirements.

- Chapter 11 summarises the conclusions that can be drawn from the work presented within this thesis and ends by describing several promising research directions for further investigation.

Chapter 2

Duplicate Tags in RFID Systems

2.1 INTRODUCTION

This chapter reviews some of the issues which surround the use of security in Radio Frequency Identification (RFID) systems. It is not intended as a complete exploration, and therefore, many tangential issues are not covered; however, a good overview of these can be found in Juels (2006). This review examines the extent to which security is relevant to the operational goal of ensuring every tag in an RFID system is unique. Unlike barcodes which usually identify a class of products, in comparison, RFID is distinguished by its ability to identify entities to the item level. However, in order to achieve this, each tag needs to be unique (Glover and Bhatt 2006). In considering how this operational principle can be invalidated, this chapter will establish a rationale for approaching security on a 'whole of system' basis.

To this end, two issues will be considered which are influential in ensuring tags in a system remain unique. The first issue is cloning, which allows the principle to be invalidated by an attacker (Bono et al. 2005). This attack is presented in various forms which illustrate that cloning is a problem which is not localised to a part of a system and is therefore difficult to prevent if security is approached at that part in isolation. The second issue is the constraints on functionality which determine how security can be implemented in various parts of a system to deal with the cloning problem. The review of constraints mainly focuses on the tag and reader as these are the more important parts of an RFID system. Having examined these issues, it will be apparent that structured analysis of at least these issues, across the whole system, is needed for the operational principle to be maintained.

2.2 CLONING

Cloning is a threat to system operations as it allows unauthorised entities to claim the privileges of authorised entities via tag credentials (Bono et al. 2005). This often occurs when the serial number of an authorised tag has been obtained and encoded onto a reprogrammable tag - there are also other ways this can occur. This unauthorised tag is in the possession of the unauthorised entity, and as the tag is a surrogate, cloning makes it challenging to distinguish entities based on tag data. Thus, security in RFID systems is needed to ensure that cloning does not prevent the achieving of a system's operational goals. The rest of this section examines the

various forms of cloning, based on where in a system attacks occur and how the attack is used.

2.2.1 TAG CLONING

This section reviews what will be referred to here as "tag cloning," which represents the more direct form of cloning as a system contains physical tags which share the same identification data. Serial-number-only systems are highly vulnerable to cloning of this form. These systems rely solely on a tag's serial number for identification purposes.

Obtaining a clone tag is no more difficult than the introduction of physical tags which contain duplicate serial numbers. A simple attack of this form is when an attacker introduces a tag which they have obtained from another system. In both systems, the tag serial number is valid, but in the latter system, the tag clashes with a serial number which is already activated. This attack is called *cross contamination* (Heydt-Benjamin et al. 2006). Similarly, the data from an Electronic Product Code (EPC) tag can be *skimmed*: read by an attacker's RFID reader and encoded onto a reprogrammable tag, to produce a clone tag (Juels 2005). As tags are validated on their serial number, these attacks are relatively easy to perform.

The following examples focus on cloning of cryptographically enabled contactless smart cards. These cards use RFID technology to identify themselves and communicate with readers, but also use cryptography for more robust identification. The review of these attacks illustrates that even RFID technology thought to be secure, when deployed in a commercial application, is little more secure than serial number only tags.

To begin with, the most commercially widespread contactless smart card, the MiFare Classic is vulnerable to cloning. The MiFare Classic has mutual authentication and data secrecy capabilities implemented using a proprietary stream cipher called Crypto-1 which relies on a 48-bit secret key. Garcia et al. (2008) have demonstrated that, by recording and studying traces from transmissions between cards and readers, the encryption algorithm and authentication protocol can be reverse engineered - revealing two attacks which enabled the secret keys of cards to be determined. With knowledge of the secret key, an attacker can read a card, clone a card, or restore a

card to a previous state. As these cards use RFID technology, contact between the attacker and target card occurs without the knowledge of the card holder, allowing the attacker to commit payment fraud surreptitiously.

Attacks against the MiFare Classic have been demonstrated in the London Oyster Card system, and the Dutch OV-Chipkaart system (Garcia et al. 2008). While the losses which could occur in these systems are financial, as the attacker obtains ticketing at no cost, when one considers that the manufacturer, NXP Semiconductors, estimated in 2008, that 200 million cards were in use, a global recall to correct the problem would come at a significant cost to the manufacturer.

Work by Courtois (2009) illustrated that with a few enhancements, the attack against the MiFare Classic can be improved so that the number of queries required is around 300. This enables the attack to be performed very quickly and suggests the ease with which the assumption that every tag in a system is unique, even when security on the tag exists, can be invalidated. As will be seen in the section on tag constraints, most low-cost passive tags have less functionality than the MiFare Classic for onboard security, so these types of attacks should be easier to perform against these tags.

Recent work by Kasper et al. (2010) on attacking the 'ID-Card,' used widely in Germany, which they report as a MiFare Classic card, reveals obvious vulnerabilities using inexpensive custom-built equipment (which costs less than 40 euros to build). The vulnerabilities enable cloning which can then enable the theft of up to 150 euros of stored value from cards in this system. While the possibility of conducting such threats is widely acknowledged in literature, because this system allows for large currency values to be stored on cards, it surprising to learn that this system does not employ additional security checks to prevent attacks from occurring.

While work on attacking the MiFare Classic by Garcia et al. (2008) has demonstrated that reverse engineering enables cloning wirelessly, work by Nohl et al. (2008) has demonstrated that card ciphers can be reverse engineered at the silicon level, using a combination of image analysis of circuits and protocol analysis. Using this type of attack, it was revealed that several weakness in the cipher exist in addition to its short key size. As the process to attack was largely automated, it could be applied to larger circuits as well; meaning future enhancements to chip size are no protection.

This work demonstrates that finding an algorithm set in hardware is within reach of an attacker, and thus, security at the protocol layers, there to prevent cloning, can be largely bypassed.

When considering the above examples, this form of cloning illustrates that the assumption that every physical tag is unique can be invalidated. When considering that more powerful RFID-enabled smartcards which use cryptography, deployed in commercial environments, fail to prevent cloning, the expectation that tags could be secured to prevent cloning in lower cost systems seems unlikely.

2.2.2 PSEUDO-CLONING

What will be referred to here as "pseudo-cloning," covers methods largely independent of a physical tag such as simulation or spoofing of an authorised identifier. As most of these attacks use simulator devices, these attacks bypass security meant to authenticate a tag.

The first example is found in the VeriChip, a commercially deployable RFID tag for use as a human implant. Halamka et al. (2006) demonstrated that a VeriChip could be cloned when an attacker uses a device to scans its data, eavesdrop on its signal, or learns its serial number. In the case of signal eavesdropping, an attacker can use the Prox Mark II - an RFID tag reading and simulation device developed by Westhues (2005), to obtain and replay VeriChip data. Alternatively, an existential cloning attack – when the tag serial number is already known by the attacker – allows the attacker's device to simulate a tag's identity without actually engaging the original tag (Halamka et al. 2006). These attacks are of the pseudo-cloning form as they occur when a replaying device is used.

Relay attacks between contactless smartcards, which use RFID technology, are another example of pseudo-cloning. These were predicted as an effective way of spoofing tag signals to a reader over long distances by Kfir and Wool (2005). In a basic relay attack, a *ghost* device simulates a card to the reader, and a *leech* device simulates a reader to the card. A channel is established between the ghost and leech, and as these have more powerful transmission capabilities than the tags or readers, these increase the range at which attacks take place. Using this type of attack, an attacker can assume the privileges of an authorised entity via the relayed tag data.

This attack was demonstrated by Kirschenbaum and Wool (2006) when they built the leech device as a portable and extended-range RFID skimmer.

Relaying can also be a way of avoiding protocol based security between the tag and reader as the next example illustrates. Relay attacks at 13.56Mhz using the ISO-14443A standard were demonstrated by Hancke (2005). To enable relaying, similar to that proposed by Kfir and Wool (2005), two devices were used: a *mole* and a *proxy*. The mole interfaced with the user's card and appeared as a valid reader, whereas the proxy appeared as a valid card to the reader, and passed instructions to the mole, which responded with tag information. In systems that do not measure added time delay – a possible way of detecting relay attacks (Reid et al. 2007) - it would be possible to circumvent cryptographic protocols using this method as they are routed through the attacker's devices.

Finally, the next example illustrates the consequences of pseudo-cloning, and what can happen when clones exist in a commercial system.

Cryptographically enabled tags have been *spoofed* - recording and replaying tag data - by a research team from John Hopkins University (Bono et al. 2005). The researchers reverse engineered the cryptographic cipher underpinning the challenge response-protocol which is used to authenticate the Texas Instruments Digital Signal Transponder (TI-DST). In this system, the key was 40-bits long (a relatively weak encryption key), and was recovered using an array of sixteen Field-Programmable Gate Array (FPGA) integrated circuits, operating in parallel in less than one hour. Given the secret key and tag serial number, this allowed the researchers to reproduce the tag's radio signals. (Bono et al. 2005).

The consequences of such attacks were demonstrated. The researchers used their simulated TI-DST to purchase petrol in the Speed Pass petrol payment system and to spoof the immobiliser authentication system of a 2005 model Ford Escape sports utility vehicle (SUV), starting it with a bare ignition key (Bono et al. 2005). In 2005, this type of tag was in use in more than 150 million vehicle immobiliser keys and also in the ExxonMobil SpeedPass system which than contained seven million units. Although the SpeedPass system employs additional fraud checking constraints in the back-end, therefore possibly detecting suspicious behaviour, a system such as the

vehicle immobiliser is relatively vulnerable, as usually there is no additional security at the ignition, except for the key. Thus, once such attacks are performed, a simulator device is now the attacking device, and security based on the tag which was targeted, is rendered ineffective.

The attacks discussed above, constitute pseudo-cloning as they result in another entity claiming the privileges of an authorised entity via tag credentials but not necessarily using a physical tag. The difference from cloning of the first form is that the original tag is usually bypassed during the attack, such as when a simulator device is used. This invalidates the assumption that every tag is unique, as actually, simulator devices are claiming to be the tag. This means tag based security is relatively ineffective when it comes to reusing the cloned data.

2.2.3 CLONING BY THEFT

What will be referred to here as "cloning by theft," allows an unauthorised entity to assume the privileges of an authorised entity by stealing a tag (not actual cloning as such) or by cloning it. However, the nature of attack is not clear from the system's perspective. In these attacks, an attacker takes advantage of systems which do not verify the associations between tags and physical entities. An attacker can remove an authorised tag from an authorised physical entity, for example, without this change being noticed by the system, until the point at which the physical entity is examined – if this occurs at all for security. In cases where tag theft occurs, even if there are physical entity authentication methods in place, such as holograms or watermarks, these do not assist an RFID system as typically these are not conveyed at the time the tag serial number enters the system (Ranasinghe and Cole 2008).

The examples now reviewed illustrate that tag or reader based security is rendered ineffective as most attacks are not definable at the tag or reader. Unless physical inspection of the entity occurs, usually solutions in the middleware such as intrusion detection are the only means of protection, as these can look for anomalies in entity behaviour by analysing the data produced by tags attached to entities.

From a system's perspective, it may be difficult to ascertain whether the identifier is coming from an authorised source. To this end, the concept of a *change-of-tag ownership* was proposed by Mirowski and Hartnett (2007). This concept

characterises how a system could perceive the use of stolen or clone tags by an unauthorised entity. As an unauthorised physical entity using an authorised tag may exhibit different behaviour from the authorised entity, it may be possible to detect misuse when analysis of RFID data is performed. Using statistical methods, Mirowski and Hartnett (2007) were the first to demonstrate that if the behaviour associated with a tag serial number does not appear to be authorised, it could be evidence to suggest that the associated physical entity is not genuine. In this case, changes in behaviour may be indicative of cloning by theft.

A different way of defining and detecting identity misuse in systems is using *synchronised secrets*. A synchronised secret is a way for a system to expose the use of duplicate tag serial numbers on different tags (Lehtonen et al. 2007c). It works on the basis that each time a tag is read, the reader writes a piece of data to the tag based on its serial number. Each time the tag is re-read, the tag conveys its serial number and its synchronised secret, and if both are valid, when compared to the records maintained by the system for the tag serial number, a new synchronised secret is written to the tag. As the authorised tag and clone tag are separate devices, the assumption is that a clone is exposed when a tag has out-of-synch data, as data is changed each time the tag serial number appears. In this case, the classification of a tag as clone is based on behaviour arising in the system being implausible. Clearly, the use of synchronised secrets must exist in the system in order for this type of implausibility to be identified. Its use, however, involves security in different parts of the system. It also relies on the original and clone being active in the system; if an attacker clones a tag and discards the original then although the clone is used, the synchronised secret would fail to detect clone tags – which is why it is classified as cloning by theft.

Similarly, the solution of a *heuristic* classification to define cloning was explored by Lehtonen et al. (2009). They characterised cloning as having occurred in a system when entity behaviour, analysed in RFID data, was deemed to be likely to have been produced by clone tags as it did not fit within a probabilistic behavioural threshold. Under the classification, this approach is detecting cloning by theft as entities may be using originals or clones.

Whether or not implausibilities are due to clones in this third form is less well defined from a system's perspective. As such, these may only become apparent if more parts of a system are included in attack detection. The detection of the attack as a cloning attack does not necessarily occur at the source of attack, such as the tag or cloning device. The detection of the attack involves the analysis of the event data generated by tag usage. Including other parts of the system in RFID security to detect these attacks illustrates that the problem of cloning is not localised to one part of a system.

To summarise, when considering the forms of cloning (tag cloning, pseudo-cloning, cloning by theft), it seems likely that threats from clones can not be defined, nor dealt with in a single part of a system. These attacks may involve an attacker claiming an identity which has associated privileges, but these claims do not always originate from a tag. Also, exposing them through detection methods, does not always clearly define what form the attack has taken, and can involve different parts of a system such as RFID data. Attacks against cryptographically enabled smart cards; which use RFID technology but are themselves more powerful than low-cost RFID, illustrate that security on the tag is largely tenuous in commercial applications. When considering these examples, it seems likely that a broader analysis of attacks in RFID systems, in the case of cloning at least, needs to be undertaken to maintain a system's operational goal of tag uniqueness. Conversely, if this class of problems was considered in isolation of the whole system, one may miss other forms of cloning which occur.

2.3 CONSTRAINTS FOR RFID AND SECURITY

In exploring why solutions on the tag have failed to prevent cloning, this section considers some of the challenges surrounding security in RFID systems. It focuses on the tag, as the tag is the cornerstone of identification in these systems. In reviewing these constraints, an overview of some of the limitations on addressing security at a single point in isolation is provided.

2.3.1 COST

Cost is a major influencing factor in achieving security in RFID systems. This section considers several cost analyses which have focussed on RFID components

such as tags. It should be noted that these analyses do not consider the variable financial aspects of cost in RFID systems such as fluctuations in currency.

In order for an RFID tag to compete with the barcode, and achieve widespread deployment, it was predicted years ago that a tag would need to cost five cents (Sarma 2001). The *five cent* tag cost model speculated that such a tag, at this cost, could be developed. However, this tag would be severely limited in terms of hardware resources. It would be a packaged tag, containing an integrated circuit (IC), 64 bits of memory, passive powering, with around 400 to 4000 logic gates and a read range of a few feet. As will be discussed, these constraints pose severe challenges for normal operations as well as for security at the tag.

If we follow the argument of Sarma (2001), it may mean that tags undergo reductions in functionality to achieve a cost of five cents; as a recent survey by ODIN Technologies, reported by news sources (Swedberg 2010a), has indicated that EPC Class-One Generation-Two tags remain at a price of around 15 cents apiece when ordered in quantities of 10,000 and 11 cents each when ordered in quantities of one million or more.

To cost five cents, most of the modifications required would be made to the IC – the most expensive part of the tag (Sarma 2001). To reduce the cost, the size of the IC needs to be reduced to less than $0.25mm^2$. There are three major contributors to the area on an IC, and consequently, these would need to be reduced: memory, logic, and power circuitry. The memory would be reduced by using read-only memory rather than non-volatile memory, and by storing only an Electronic Product Code (EPC) - a tag's serial number. The EPC is used as an index into a database where additional information about the entity can be found. The logic on the chip would be minimised by using efficient anti-collision protocols such as the Tree Walking protocol (Glover and Bhatt 2006). Finally, the power circuitry would be minimised, a 'synergistic effect' of minimising the other areas of the tag (Sarma 2001). Considering these limitations, this costing model places severe limitations on what would normally be operationally achievable for tags, let alone, what could hoped to be achieved for security.

As will be discussed, the implications of constraining the cost of a tag, and hence its capabilities, mean security on the tag is weak, and this has consequences for the entire system. This was briefly alluded to in the previous section when the Texas Instruments Digital Signal Transponder (TI-DST) was mentioned (Bono et al. 2005) - if security cannot be addressed at the tag, then additional security measures like fraud checking need to be put in place. Such solutions are themselves, not without system implications. This was illustrated when Mirowski and Hartnett (2007) demonstrated that incorrect classification of attacks causes false positive alarms, meaning entities may be prevented from accessing the system. Thus, security in RFID systems involves system-wide considerations.

The concept of low-cost tags was largely formalised by the Auto-ID Centre when they developed EPC technology and a related class hierarchy for RFID labels (Ranasinghe et al. 2004). This hierarchy specifies the amount of functionality prescribed to tags that conform to this cost model. Class-One and Class Two represent low-cost tags and contain simple EPC read-only labels and read-write memory, respectively. In 2004, Class-One labels consisted of around 1000 to 4000 logic gates while Class Two labels contain several thousand more logic gates. The IC memory had a few hundred bits available for data storage. The memory use between these two classes was distinguished between read only memory for Class-One tags, whereas Class Two tags have read-write memory. The logic on board these tags was designed to only execute reader commands and implement an anti-collision scheme. In terms of power, at the time it was estimated 150 microwatts of power would be needed to operate such a tag. As this specification was developed with the intent of standardising basic tag operations, it seems likely that any security would be constrained by this specification. Thus, proposed security features may need to be backwards compatible.

For cryptography, the predominant tag based solution to cloning attacks (Juels 2006), the implication of constraints influenced by cost, are essentially this: given the limitations on the number of logic gates, private key cryptosystems such as the Advanced Encryption Standard (AES) are not suitable in commercial implementations as AES typically requires upwards of 20,000 gates (Ranasinghe et al. 2004). As the number of gates available to low-cost tags, which is less than 4000, is far below this requirement it seems likely that cryptography would have to be less

secure in a commercial implementation. The John Hopkins example described above (Bono et al. 2005) demonstrates this, as the asymmetric nature of computing resources available to the attacker and on the tag means that tags will nearly always be at a security disadvantage.

When considering these costing models, the constraints which exist on low-cost tags are rigid. The development of security solutions, would come as an addition to the normal operating conditions of tags, and would therefore be subject to not only the constraints which are imposed on the whole tag, but also the constraints imposed on competition for resources between the security solution and the tag's normal operations.

2.3.2 FREQUENCY

The radio frequencies at which tags and readers operate is another constraint on achieving security in RFID systems at the tags and readers. The regulations surrounding each frequency place constraints on power emissions, which in turn influence: read distance, transaction time and anti-collision methods. These issues, amongst many others surrounding the use of certain frequencies for RFID security are now briefly discussed.

Most RFID systems operate in the Industrial, Scientific and Medical (ISM) bands designated by the ITU (Ranasinghe et al. 2004). The most commonly used High-Frequency (HF) ISM band in Europe and America is centred at 13.56 MHz and the Ultra High-Frequency (UHF) band in the US is 902-928 MHz. Each frequency band places regulations on the isotropic radiated power at certain distances; thus, tags have different types of constraints according to frequency.

For passive tags these place severe restrictions on what can feasibly be achieved on the tag. These regulations place an upper limit on the power available at a given tag distance from a reader (Ranasinghe et al. 2004) and these place limits on the type of security scheme and cryptographic hardware. As power emissions are limited, if cryptographic hardware is used, it will use the available power or it will require additional power.

As it stands, tags which operate at 13.56 MHz have typical reading ranges of 30 centimetres (cm) to 50cm; conversely the United States regulations allowing the

longest communication, 902-928 Megahertz (MHz), passive Ultra High Frequency (UHF) RFID tags have reading distances of around 3m to 5m. Any onboard addition, such as cryptography, therefore, may be detrimental to these estimated read distances. If cryptography uses the available power sent from the reader, it will severely diminish the tag reading distances and degrade the performance of the whole RFID system (Ranasinghe et al. 2004). When considering this consequence, diminished read range may place low-cost tags on par with barcodes, making the perceived benefit of longer read distances less attractive. Conversely, the depth at which tags embedded in objects could be read may be influenced; tagged objects inside packages may not be as readable requiring the packages to be unpacked impacting on a tag's non-line of sight feature.

Frequency also places limitations on communication time (Ranasinghe et al. 2004). In the United States, for example, UHF regulations for frequency hopping specify a maximum time limit of 400 milliseconds (ms) on the use of a frequency channel. This means a tag cannot be assumed to be in continuous communications across a frequency hop, further limiting security transaction time. For readers, the ability to read a minimum number of tags within a given time period - such as 300 tags per second - may be diminished as any increase in transaction time, which would come from security protocols, will inevitably reduce maximum read rate. Consequently, this may mean that fewer tagged entities are identified within a time period, in which case, the total number of entities in the whole system may take longer to identify.

There are also limitations on bandwidth. According to work done by Sarma et al. (2003), a tag at 13.56 MHz must use far less signalling between readers and tags when compared to what can be achieved at the 915 MHz. Consequently, implementations for anti-collision protocols vary accordingly, to account for this lack of bandwidth. For example, Aloha-based anti-collision algorithms are more common in systems that operate in the 13.56 MHz band, whereas deterministic anti-collision algorithms are more common in the 915 MHz band. Similarly, protocols to secure the tag at 13.56 MHz must use far less signalling from reader-to-tag than at 915 MHz. Thus, the decision to use a certain frequency has ramifications for what can be achieved between the tag and reader in terms of security.

To summarise, when considering the examples above, it appears likely that cost and radio frequency are factors which influence the development of RFID security on components. Cost imposes constraints more severely on those tags which are situated at the bottom end of costing models. The five cent tag clearly represents a low cost tag, but with little capability for supporting tag based security. Frequency imposes system design constraints as well as a security design constraints. The choice of frequency affects the read range, read rate, and bandwidth – all of which influence the operation of the whole system. The example used was that of entity enumeration in systems: increases in frequency resource usage, because of cryptography, will be detrimental to the basic goals of enumerating tagged entities over long distances, in the hundreds per second, and through multiple layers of product packaging.

2.4 SUMMARY

This chapter began with an exploration of cloning to illustrate that forms of this attack relate to different parts of an RFID system and then it considered the constraints which make it difficult to protect systems from cloning. This review has suggested the need to consider the 'whole system' in analysing the security requirements of a system to account for these issues.

Cloning attacks invalidate the operating principle that every tag in a system is unique meaning unauthorised entities can be validated as authorised. Cloning is constituted differently in various parts of the system. It is easy to define at the tag but more challenging when it occurs from a cloning device which may be relaying a signal through a ghost and leech. Relaying attacks illustrate that even tags with security are vulnerable to cloning, whereas attacks against RFID-enabled smart cards - which are generally more powerful computationally than low-cost tags - illustrate the challenges of implementing strong security in commercial applications.

The need to consider the 'whole system' in defending against cloning was made more apparent when constraints in the system were reviewed. The 'synergistic effects' (Sarma 2001) expected due to cost and frequency goals suggests that implementing security involves tradeoffs. When developing low-cost tags: integrated circuit (IC) size influences logic gates, memory, and powering; and the

choice of frequency influences read range, read rate, and available bandwidth. The introduction of cryptography in places like the tag, complicate these linkages. One consequence is suggested as reduced read range as cryptography on passive tags will require more power for logic gates. Whether this is likely to be feasible is not clear as commercial implementations of standard algorithms require many more gates than those which are currently available for a projected cost of around five cents.

This all suggests that consideration should be given to the 'whole system' when considering how to respond to threats like cloning. The achievement of sufficiently strong security, within the constraints, and in relation to specific implementations, is challenging. To this end, the next chapter reviews previous work which has considered how analysis can assist in examining the security requirements of RFID systems.

Chapter 3

Current RFID Security Solutions

3.1 INTRODUCTION

This chapter reviews previous work which has analysed security in RFID systems. It starts with a review of what broadly constitutes 'the system' and uses this as a basis for examining the appropriateness of existing systems analysis approaches to security. The review of system analysis approaches focuses on the structures which make up the system, as the previous chapter has already identified some of the components which exist. This serves as the basis for examining how, and where in the system, security analysis has been focussed. As the previous chapter has suggested, security relates to the 'whole system', therefore, this review will attempt to establish how much of the system is captured in these approaches.

3.2 RFID SYSTEM ANALYSIS APPROACHES

To understand the place of security in RFID systems, this section begins by reviewing what is considered as 'the system'. To this end, systematisations of RFID components and other elements are reviewed along an architectural basis, that is, identifying the internal and external system boundaries which organise elements.

Whether such systematisations are actually appropriate is contentious amongst some researchers - Glover and Bhatt (2006) have suggested no single universal RFID system architecture exists given the wide variety of application environments for RFID. As will be illustrated, the following architectural perspectives and representations could be perceived as subjective, having been developed by various authors based on their own set of beliefs. However, to avoid potential biases in this review, a number of perspectives are examined, and this section then attempts to extract properties which are common to systems from a number of these examples.

3.2.1 RFID SYSTEMATISATIONS

Two categories of RFID systematisation are reviewed: assessments and models. Assessments are characterised by their textual description of system features, whereas models are focused on graphical abstractions. This distinction is used to illustrate the advantages imparted by each approach in its description of systems information. As will be seen, the latter category not only imparts information concisely, it does so in a means by which the architectures are actually reflected in

the reporting approach. From each of the categories emerges the properties of what broadly constitutes the architecture of 'the system'.

The concept *hierarchy* is a characteristic of some RFID systematisations. Hierarchy appears to emerge from the organisation and information flows of the three major RFID components (Garfinkel and Holtzman 2005). Usually *tags* are the lowest element in the hierarchy, above which are *readers*, and then *databases*. Sometimes these components have been further decomposed to include the *antenna* and *middleware* components, however, usually these components are accepted as parts of the larger components. Usually, tags are below the readers in the system as these are attached to more entities and the readers are there to aggregate identification.

The information which flows from the tag, when its data is obtained by a reader, is transferred into a database. In this ordering of information flow, the tag is subordinate to the reader as it is the reader which initiates commands, to which the tag which responds. This instantiation of hierarchy, albeit a temporal one, has been prevalent in Electronic Product Code (EPC) technology (Ranasinghe et al. 2008). However, it is a characteristic that seems likely to apply to most RFID systems, as most systems function on this basis (Glover and Bhatt 2006). Thus, the organisation of components and the information flow between components forms a basic architectural property of RFID systems – a hierarchy.

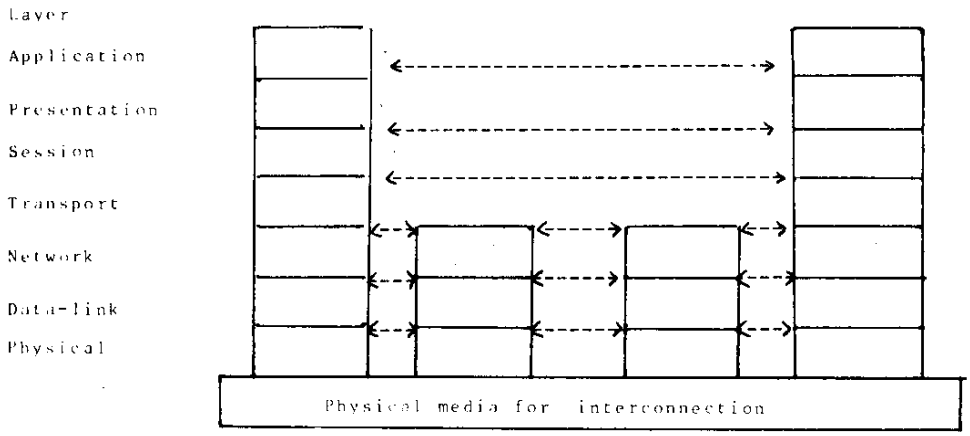

Figure 1 - Original (low-quality) depiction of the OSI reference model
The model depicts the seven layers of the model and illustrates how systems are interconnected through a common layer. (Zimmermann 1980)

What constitutes an RFID system's *hierarchy* has been extended by the introduction of the organisation of components into a series of *layers*. Layers have been used to organise RFID components into distinct categories; from the physical environment through to the RFID technology, to the information goals of the system. As will be seen, what constitutes these layers varies amongst models.

The use of *layers* as a systems concept seems to stem from the similarities between RFID systems and communication systems. A well-known communication model, the Open System Interconnection (OSI) Reference Model (Zimmermann 1980), illustrated in Figure 1, uses layers to organise communication elements. Consequently, various authors in RFID have used OSI layers for describing the architecture of these systems.

On the basis that RFID systems are a type of communication system, Shepard (2005) systematised RFID components using four layers of the OSI model. The model he proposed uses: *physical layer* (layer one); *data link layer* (layer two); *presentation layer* (layer six); and *application layer* (layer seven). The model does not use the other OSI layers as things like routing and congestion control are irrelevant to the RFID systems.

Table 1 illustrates the use of OSI layers in RFID systems and includes the descriptions provided by Shepard (2005) in an additional column to show why they are used.

Table 1 – OSI Layers and ISO RFID standards
The model uses the concept of layers and their use is justified by similarities between the OSI layers and various RFID technology standards. (Shepard 2005)

OSI Layer	Associated Standards	Description
Layer 1 (Physical)	ISO 14443-2 for Type A, B devices ISO 7816-2 for contact card implementations	The physical interface between tag and reader. Bit rate, electrical signal representation.
Layer 2 (Data Link)	ISO 14443-4 ISO 7816-3 for contact card implementations	Address management using the Channel Identifier (CID) field, transmission of sequential data blocks, link control, and anti-collision.
Layers 6 and 7 (Presentation and Application)	ISO 7816-4 ISO 7816-7 Various vendor-specific proprietary protocols	The presentation and encoding of data such as encrypted data between tag and reader. The onboard tag or reader software.

Other authors have also made frequent use of the OSI layers, however, the common use is only to describe the RFID technology. Avoine and Oechslin (2005) have used the concept of OSI layers to organise RFID components for traceability threats. Figure 2 shows the *communication model* they proposed, which uses three layers: *physical layer*, *communication layer*, and *application layer*. The *physical layer* captures the RFID air interface elements such as frequency, modulation of transmission and data encoding. The *communication layer* contains the protocols which make possible interaction between a tag and a reader such as anti-collision protocols. Finally, the *application layer* represents user defined information relating to the entity. It is worth noting the model was applied to the analysis of a privacy

problem called *traceability*. Traceability is an attack which allows an attacker to monitor the whereabouts of a particular person through their RFID tags (Avoine and Oechslin 2005). As the model does not make explicit that it is restricted to only privacy considerations it has been reviewed in this section.

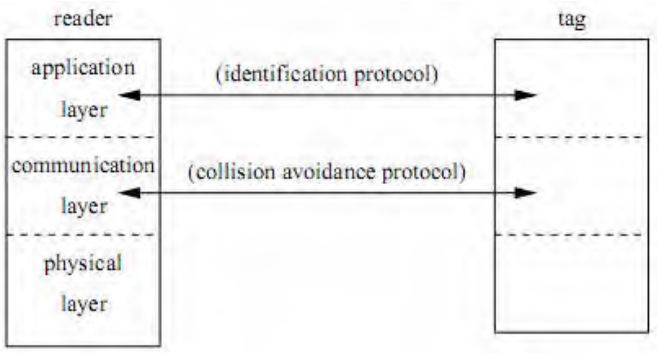

Figure 2 - Communication model
This is an example of an early attempt to use a model for analysis of security in RFID systems. It systematises RFID using three OSI layers, and is used as a basis for examining traceability threats. (Avoine and Oechslin 2005)

This model appears to represent the first known example of analysis of security in RFID systems over layers. In the next section, it will be seen that this is an increasing trend amongst various researchers. However, the above example is limited to security only at the tags and readers.

As RFID systems are used within physical environments, some researchers have introduced layers beyond the RFID technology itself. In Figure 3, Mitrokotsa et al. (2008, 2009) introduced the concept of a *physical layer* that captures physical components which pertain to the RFID system. The *communication layer* and *data link layer* are reused from OSI-based models, like those seen above. However, in their model these are represented as a combined layer, the *network-transport layer*. This combined layer includes all the protocols responsible for tag and reader interaction. The concept of an *application layer* has been extended in this model. It captures any enterprise level application. Finally, a *strategic layer* is introduced which contains the information goals of a system's owner e.g. a company. When considering this example, it is apparent that the notion of what constitutes an RFID system can include more than just the RFID technology.

When considering previous examples, it is worth pointing out that the model by Mitrokotsa et al. (2008, 2009), is distinguishable on the basis of vertical demarcations at each layer. These demarcations organise elements into *object classes*. Figure 3 shows that, for example, tag and reader components, of which there could be many types, are organised into two separate classes: reader hardware (Reader HW), and RFID tags. These vertical demarcations are a way of including more detail at each layer.

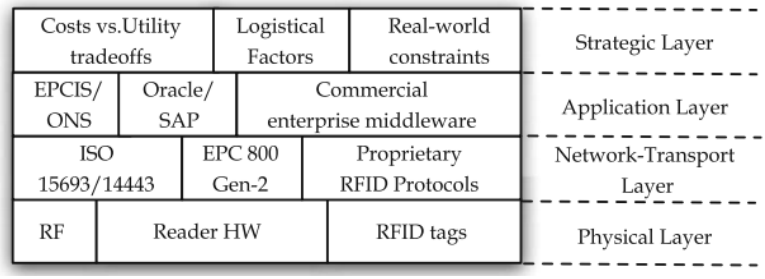

Figure 3 –Layers beyond the RFID technology
The use of a physical layer and the strategic layer allow a broader view to be taken of what constitutes the system. The use of vertical demarcations enables elements at a layer to be grouped. (Mitrokotsa et al. 2008, 2009)

Conversely, a taxonomic representation has been proposed by Hassan and Chatterjee (2006). The taxonomy organises RFID system components as members of four object classes. One class, the physical class, is illustrated in Figure 4. The *usage* class organises systems into two roles: *monitoring* systems or *authorisation* systems. The *frequency* class organises communication protocols. The *physical* class organises tags and readers based on hardware configuration. Finally, the *data* class organises the enterprise level services performed by the system. In this taxonomy the organisation of elements, with higher level elements having precedence over lower elements, appears to form a hierarchy. As natural divisions are formed between classes and sub-classes in this taxonomy, and as the branches extend from a root node and sub-branches are introduced, it appears as though this taxonomy achieves a similar vertical separation of concepts to the vertical demarcations which appear in the model presented by Mitrokotsa et al. (2008, 2009).

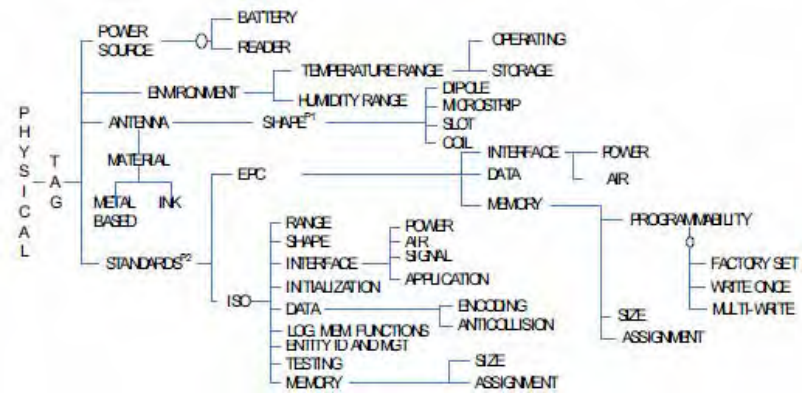

Figure 4 – The physical class
The taxonomy decomposes system elements into various classes to identify their constituent elements, but does not signify where in the system these elements are located. (Hassan and Chatterjee 2006)

When considering previous examples, Glover and Bhatt (2006) have expanded the systematisation of RFID by including *middleware*. In their model, illustrated in Figure 5, *middleware* is a concept which is used to capture anything that supplies data to the enterprise management applications which are used by clients for information processing. Sensors - such as RFID readers, as these detect the presence of tags - are responsible for collecting sensory data such as temperature or tag data. This data is then transferred into device drivers which coordinate the transfer of all data into the middleware. An event database located in the middleware is responsible for aggregating and transforming the data, and then preparing it for use by applications. The interfaces in the middleware seem to form layers around which components of the middleware are grouped. If one were to rotate the model on its side, the arrangement of these demarcations would be horizontal like Mitrokotsa et al. (2008, 2009) layer's rather than vertical.

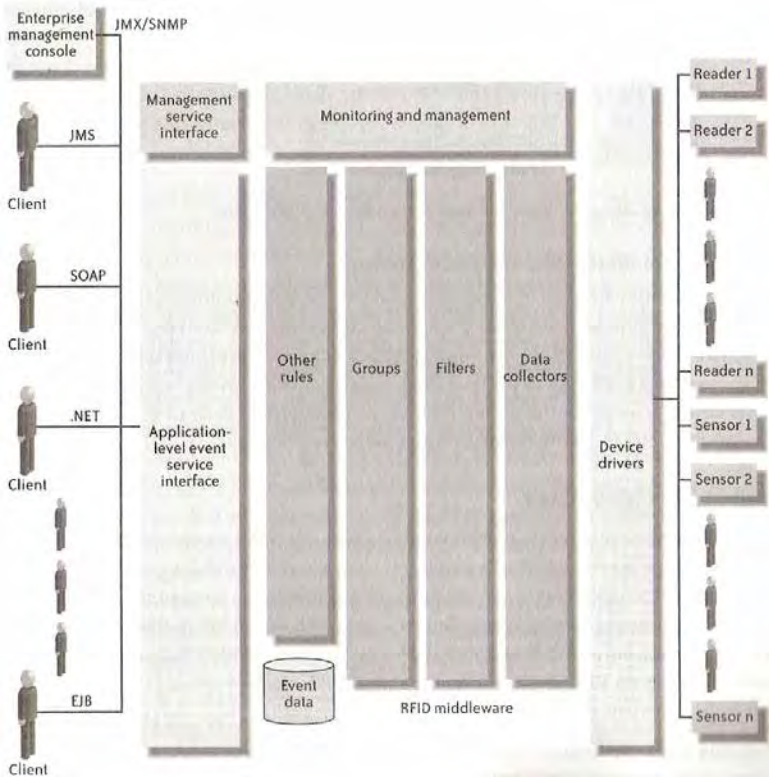

Figure 5 - Conceptual architecture for an RFID middleware
The model illustrates that middleware can be decomposed into layers thereby extending the layers from sensors to the clients which operate at the enterprise level. (Glover and Bhatt 2006)

Another systematisation of RFID was proposed by Glover and Bhatt (2006). Figure 6 illustrates a model which is based on an RFID application of a *retail-store.* It adds contextual information such as where in the enterprise merchandise is stored for sale (in addition to it being sourced from business partners). There are merchandise items which the retail store is monitoring via RFID technology. These are confined to shelves and checkout lanes, also where the physical tags and physical readers are located. RFID middleware software modules are depicted and these manage the readers. Edge applications represent any enterprise application that has components operating in the store, such as point of sales (POS) systems. The RFID Edge Information Service stores RFID events and related data in the system. Externally, the retail store is connected to a data centre whereby, enterprise level data processing - one assumes on the RFID data feed - is occurring. Thus, it shares a hierarchical organisation and separation of elements into groups which constitutes a layered model.

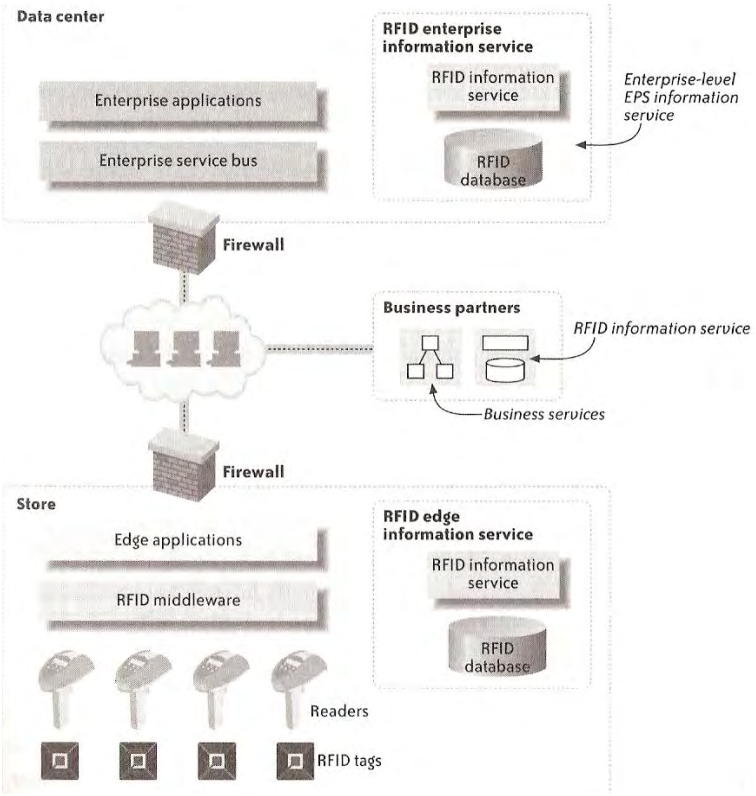

Figure 6 – RFID system in a retail application environment
This model includes the elements which are external to the RFID system such as a *store* and *data centre*. (Glover and Bhatt 2006)

To summarise, when considering the above examples, *hierarchies* and *layers* appear as common architectural properties used to systematise RFID components, but some models have also used these for organising other elements. Some models achieved this wider view by including the application environment and the company which operates in the system. The addition of these properties illustrates that the application environment is relevant to the discussion on what constitutes 'the system'.

3.3 RFID SECURITY ANALYSIS APPROACHES

This section reviews approaches to security analysis in RFID which fall into the two general categories of assessments or models from the previous section. As with the system descriptions, assessments have focused on describing the elements of the problem domain, whereas models have focused on graphical representations. This

review builds upon the above systematisations of RFID as a basis for evaluating how closely security is integrated into the context of RFID systems. Reference will be made to the previous chapter, which examined cloning attacks and system constraints, and, along with the security properties which emerge from these approaches; a view will be presented on the appropriateness of existing approaches to examining security on a 'whole of system' basis.

3.3.1 SECURITY ASSESSMENTS

Security assessments have focussed on elucidating the properties of systems and security and (loosely) relating these to one another in an attempt to illustrate the feasibility of security requirements.

Privacy concerns in systems have been considered by Ranasinghe et al. (2004). A set of constraints for low-cost RFID was described, these included: manufacturing costs, capabilities of integrated circuits, frequencies and regulations. In the case of a spoofing attack, whereby the behaviour of a tag is imitated by an electronic device, this threat was seen to arise in part due to the lack of authentication on the tag. In addressing this threat, consideration was given to the constraints on the tag e.g. the number of logic gates potentially available for cryptography. This solution has focussed largely on the tag and it appears to lack detail on how security operates in other parts of the system and therefore lacks a system wide perspective.

When system-wide constraints were explored by Sarma et al. (2003) it was shown that these have implications for security. Consideration needs to be given to the system constraints, which include: transceiver-transponder coupling and communication, data coding, modulation, anti-collision, frequency and regulation. This means, for example, to prevent eavesdropping within the interrogation zone of a reader, tags may need to encrypt their data using a random nonce to prevent tracking; however, supporting strong public key cryptography is beyond the resources of low-cost tags. This approach illustrates an assessment based approach to analysis, as security is considered in relation to system constraints, requiring some understanding of the interconnectedness of system components. However, when considering this work it is apparent that consideration was not given to parts of the system beyond the tags and readers.

The major shortcoming in the above examples appears to be the lack of system architectures to organise elements. While the above examples have given some consideration to how constraints influence system design, and in turn influence the feasibility of security, including system structures would assist in distinguishing to which layers these effects are relevant. This may assist in realising at which layer security should be situated to deal with various attacks.

Conversely, some approaches have focused on fewer systems facets when deliberating security requirements for systems.

Peris-Lopez et al. (2006) considered various security threats and solutions, and an overview of RFID systems was given brief treatment when various components were considered. While system characteristics were considered, these were not really used to provide a robust framework in discussions centred on threats and solutions. That is, proposed solutions were discussed, such as the Kill command, Faraday cage approach and Blocker tag; however, these were not considered in a system context. Consequently, it seems likely that it would be difficult to determine the feasibility of security in different parts of a system through this assessment.

Thompson et al. (2006) considered a characterisation of RFID threats. They used the STRIDE (Howard and LeBlanc 2003) method of threat analysis, which is an abbreviation for: spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege. Consideration was given to various threats within two of these criteria, and these are now briefly mentioned. *Spoofing identity* includes: scanning RFID tags using an unauthorised reader. *Tampering with data* includes: an attacker modifying tag data such as modifying a high-priced item's EPC to be the EPC of a lower costing item. When considering these examples, the analysis made possible by this approach is relatively limited. Consideration to the associated system constraints, for example, is not related to how attacks arise. Thus, while the threats are considered, the constraints these would be influenced by, which determine if they are in fact feasible for actual systems, does not appear to have been considered.

Again, when considering the above examples of security analysis, the major shortcoming which is apparent is the lack of structure to the analysis. Each analysis

has considered different parts of security in relation to systems, usually, but not the structures which are apparent in system descriptions. For example, Thompson, et al. (2006) have listed *information disclosure threats* such as a retailer receiving a tagged pallet but then claiming the pallet was never received. However, this analysis fails to take into account in which parts of the RFID system repudiation occurs. For example, if a trace of the pallet travelling through a supply chain had been formed, and the delivery company had a record of delivering the pallet, then it seems likely such a threat would not be valid. Of course, this all depends on system context. In comparison to system assessments, the above examples appear to lack any noticeable context using established system architectures.

3.3.2 MODEL BASED ANALYSIS APPROACHES

Model based analysis approaches are an alternative when compared to the assessment based approaches discussed above. In each of the examples now discussed, a model has been constructed which represents the relationships in RFID and security, and these form the basis for a security proponent to deliberate about during security analysis.

Spiekermann and Ziekow (2005) considered various privacy threats, attack feasibility, and solutions. The attack tree threat method (Schneier 1999) was used to structure attacks around a central attack goal. One attack goal considered is for an attacker to identify a person's belongings based on the tags attached to those belongings. This could occur if an attacker was intent on determining, for example, the contents of a person's home prior to targeting the home for burglary. The intermediate steps in achieving this goal were organised into attack tree. In this case, the attacker needs to obtain and interpret EPC data and prior to this, exploit reader-to-tag communications, or read the tag data off some tags. These elements constitute the attack tree illustrated in Figure 7.

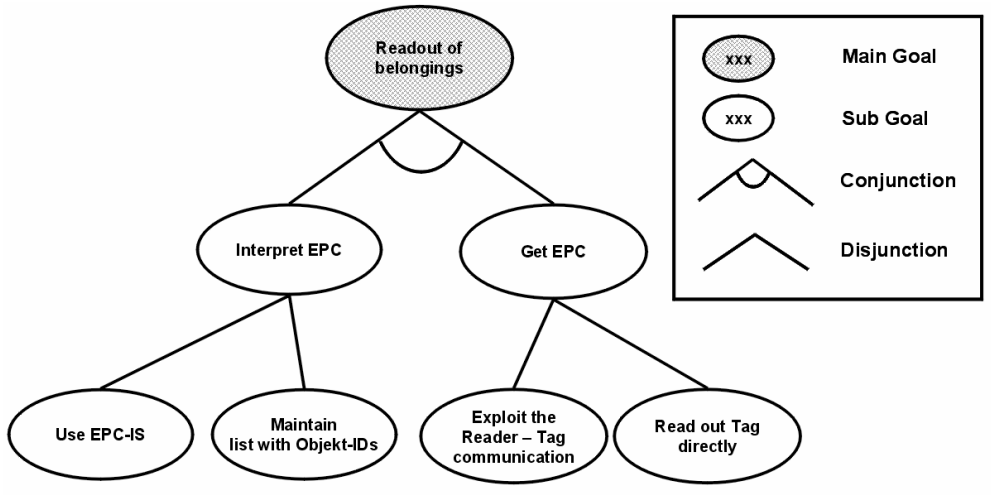

Figure 7 - Attack Tree for assessing objects
This model organises the various ways an attacker may scan individual items to determine an individual's belongings via attached tags. (Spiekermann and Ziekow 2005)

In considering the examples provided in the various attack trees by Spiekermann and Ziekow (2005), clearly there are limitations on what can be achieved in each attack sequence, due to constraints in systems. For example, to determine what an EPC tag is identifying, an attacker would need to consult an EPC Information Service (EPC-IS) database. Such limitations are discussed; however, the major limitation is that the discussion is not aligned with the attack tree. As constraints pertain to nodes in a tree, and the attack tree does not depict this information it is not explicit whether attacks are feasible. Moreover, in considering possible solutions, again, some constraints are discussed, but the influence of constraints on these, across the whole attack tree is not clear. For example, in one case, a hash-lock is proposed as a potential solution – but no depiction is given of the implications the use of this solution would have on system performance – a clear addition to tag functionality which, when considered with the 'synergistic effects' discussed by Sarma et al. (2001), would have implications for different parts of a system.

The next examples of security analysis approaches have introduced security models which enable deriving security requirements for systems. Each introduces a means of classifying threats and solutions, and subsequently, deriving an indication of the amount or type of security to employ.

Mitrokotsa et al. (2008, 2009) classified threats by system layer, and solutions were related to each threat at a layer. The physical layer comprises the physical RFID

devices and these are vulnerable to physical modification by an attacker. For example, a tag could be removed from its entity, physically damaged, or destroyed using the Kill command. Defences to physical layer attacks can include: increased physical security; enhanced attachment of tags to entities; or stronger kill passwords. The relationship of the attack to the solution is generally suggested as constrained to a single layer, and appears as such in the model seen in Figure 8.

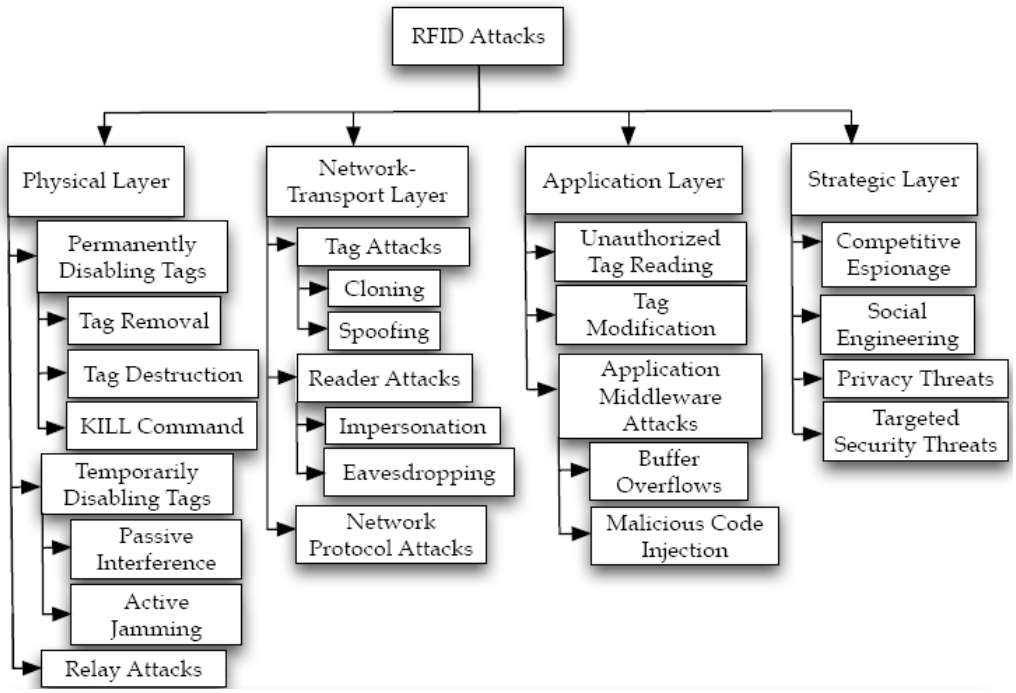

Figure 8 - Classification of RFID attacks at layers
The use of layers indicates where in a system threats and solutions co-exist. (Mitrokotsa et al. 2008)

Some attacks are suggested as occurring across multiple layers, such as replay attacks. As an attacker may record a signal, they may reuse it at a later time in order to gain physical access to a part of the system. Some suggested solutions include the use of timestamps, challenge response cryptography and radio frequency shielding.

When considering the above example, it seems likely that many more attacks would have influences across the layers. Tag removal, whilst depicted at the physical layer, would mean that a denial of service attack occurs at the strategic layer, as the entity monitored by the tag is no longer producing data in the system for it be identified. Thus, localising threats and solutions limits the capacity to represent the relationships which exist between attacks and solutions throughout the system.

Mitrokotsa et al. (2010) introduced a layered security model which captures threats and solutions at layers: RFID edge hardware; communication; back-end. The RFID edge hardware layer contains the tags and readers. The communication layer contains the radio link between tags and readers. The back-end layer contains the middleware components like databases and web servers. Elements of each layer are subdivided into three security properties: confidentiality, integrity and availability. Another characteristic of the model is that each classification has assigned values for various attributes: associated damage caused by threat; cost to implement threat; type of tags most vulnerable to threat; possible countermeasures; and associated costs.

	Attack	Potential Damage	Attack Cost*	Class of Tag	Solution - Cost*
Confidentiality	**Side Channel Attacks**	- Extract information (i.e. cryptographic keys).	**H**	Low Cost Tags	- Use of tamper resistant tags. **(H)** - Limit electromagnetic emissions. **(M)** - Increase complexity of the circuit. **(H)**
Integrity	**Physical Data Modification**	- Altering data stored on tag memory.	**H**	Low Cost Tags	- Memory protection. **(M)** - Secure cryptographic protocols. **(M)**
	Impersonation	- Supplant legitimate tags. - Elicit sensitive information. - Gain unauthorized access to services.	**M**	Low Cost Tags	- Use of tamper resistant tags (i.e. Physical Unclonable Function (PUF)). **(H)** -Memory protection mechanisms. - Physical protection (against tag swapping). **(H)** - Use of encryption techniques. **(M)**
Availability	**Permanently Disabling Edge Hardware**	- Avoid identification. -Untraceability of tagged objects.	**L**	High/Low Cost Tags	- Rugged, flexible tags. **(M)** - Increased physical security. **(H)** - Efficient key management (regarding command abuse). **(M)**
	Temporarily Disabling Edge Hardware	- Avoid identification. -Untraceability of tagged objects.	**M**	High/Low Cost Tags	- Have limited number of unsuccessful reads. **(L)** - Store both the old and the potential new key or pseudonym values. **(M)**

*Cost: **H** high, **M** medium, **L** low.

Figure 9 – Edge Hardware Layer threats and countermeasures
This figure organises threats and solutions at this layer according to security principles. (Mitrokotsa et al. 2010)

There appears to be a number of opportunities for improving upon this model for it to be used for 'whole of system' analysis. The model depicts a strong correspondence between attacks and solutions at a single layer and a single security principle. For example, a side channel attack is dealt with using a tamper resistant tag; limiting electromagnetic emissions; or increasing a circuit's complexity. These attract a medium to high cost of defence, and generally target low cost tags. However, the classification does not appear to indicate what influence this attack has on other attacks at the same layer or layers above it. A side-channel analysis, for example, can be performed when the attacker undertakes simple power analysis

(SPA) attack to reveal the kill password of UHF tags, and once they have the kill password, the attacker may disable a tag. Consequently this would cause a denial of service attack at the same layer, as the entity no longer has a valid tag attached to it allowing it entry into the system. This would also mean a database in the higher layers may not be able to resolve the location of the entity as it will no longer generate a history of movement in the system. Thus, it appears that the organisation of an attack, and subsequent solutions at the same layer, could be improved to account for these interrelationships which arise when an attack is perpetrated.

Following this, as attacks can have interrelationships throughout a system like those described above; the model could be improved on the point of how it assigns values to each classification. Potential damage for example may vary according to the other attacks involved in an attack, as illustrated above. Performing an SPA may not involve any damage; however, its combination with the kill password would have a potentially high damage rating. Thus, the derivation of these values may vary as different combinations of attacks are used. Similarly, the derivation of values for solutions, when coupled at the same layer would benefit from the same improvement. In the case of impersonation attacks, suggested solutions include: tamper resistant tags; memory protection mechanisms; or encryption. It may be the case that dealing with this threat directly at this layer involves the associated medium to high costs; however, this overlooks the ability to deal with impersonation attacks using solutions at the higher layers. Intrusion detection (Mirowski and Hartnett 2007) or synchronised secrets (Lehtonen et al. 2007c) are two solutions which could assist in thwarting this attack and for very little cost to the tag or system; however, these are not depicted nor feasibly represented in the model as these solutions engage elements throughout a whole RFID system.

Finally, by classifying attacks and solutions using *security principles* at each layer, this model appears to suggest that a threat is unlikely to impact on other security principles at the same layer or at different layers in the system. It may be true that a *replay attack* is an integrity attack at the communication layer, for example. However, if RFID data, which is replayed, goes on to be inserted into a database, which is at a higher layer in the system, it will invalidate the assumptions of the database – therefore, causing an integrity attack at a higher layer. Thus, as security

principles are not always confined to the layer at which an attack occurs, there is an opportunity to improve on this characteristic.

When considering the above example, it seems likely that dividing security into three main layers which broadly constitute RFID technology, and confining elements to security principles and categories, could be improved upon to take into account the synergistic effects and interrelationships which arise when elements interact across layers.

The next analysis of security is performed using a framework which represents several system properties to derive a security classification. When compared to the previous examples, it makes actual system classifications.

Rotter (2008) has introduced a framework, illustrated in Figure 10, to assess privacy and security in RFID systems. The framework has been used to derive a classification of various domain risks using three criteria: a system's deployment range; the link between the RFID tag and identity-related data; and the domain security demands. Using these properties, various systems have been classified, and their security demands determined. Some classifications appear to overlook relevant properties which are influential in determining the security requirements of a system. These are now discussed in conjunction with the security classification suggested by the framework.

To begin with, according to Rotter (2008) "most industry applications" demand "medium to low security" as these systems are "closed systems" which have a low link between an RFID tag and identity-related data. This appears to assume that nobody outside the system can attack. However, attacks such as cross contamination imply that cloning attacks can be transferred between systems (Heydt-Benjamin et al. 2006), invalidating the assumption that a closed system is safe. An attacker within the organisation, using a cross contamination attack, could obtain the privileges of another individual in the system, thereby invalidating this classification.

Moreover, the MiFare Classic card which is used in some very large closed systems does have a very strong link to individuals. The London Oyster card system is used by individuals for transportation ticketing (Garcia et al. 2008). Given the vulnerabilities of the MiFare Classic, it seems likely that this type of system, which

is closed, but has a strong link to individuals, should have a very strong demand for security – and not a medium to low demand for security. When consideration is also given to the widespread use of the MiFare Classic around the world, which has been estimated at over 200 million cards, it seems likely that there would be a very high number of systems at risk of attack.

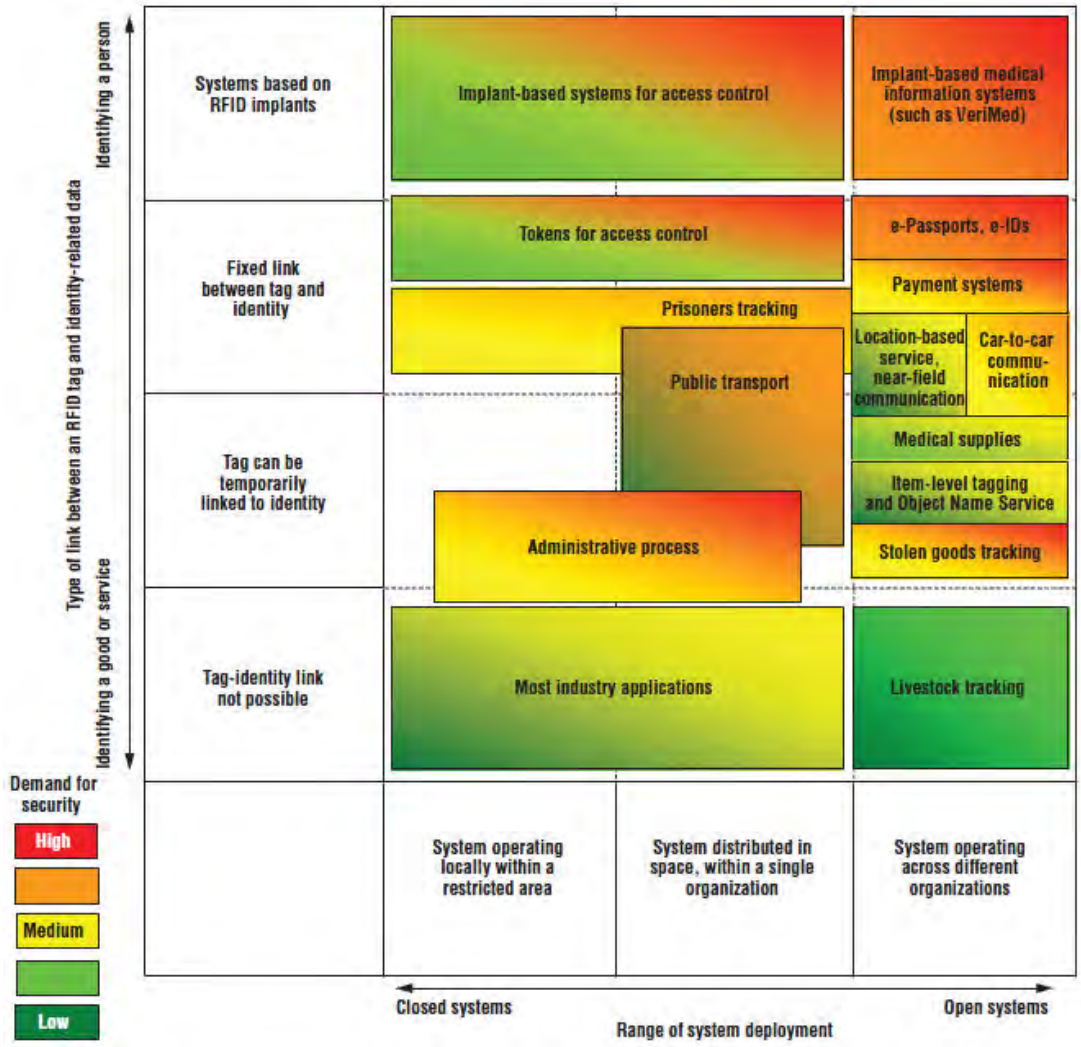

Figure 10 - Privacy and security risk assessment framework
It represents security as a trade-off between range of system deployment, and the link between tags and personal identity. (Rotter 2008)

Similarly, medical supply systems, item-level tagging and Object Name Service (ONS) systems – although classified as more "open systems" and "systems which establish a temporary link to identity," are also rated as having "low to medium" demand for security. This classification, made possible by the framework, seems to overlook the value of the entity attached to the tag. The necessity to make this consideration can be seen in the following example. As previously mentioned,

GlaxoSmithKline has implemented an RFID system to monitor the *Trizivr* Human Immunodeficiency Virus (HIV) drug bottle via tags at the item-level (O'Connor 2006). It seems likely that if an attacker could introduce counterfeit Trizivir into the system on the basis of performing a cloning attack to assign a valid identity to counterfeit products, there is a potential risk that users of the drug fall victim to inferior quality Trizivir. Thus, consideration in medical supply systems or item-level tagging systems needs to give consideration to the entity, not just the link to privacy. Moreover, as the Trizivir system is using EPC tags, when considering the analysis of surreptitious scanning of personal belongings for EPC tags by Spiekermann and Ziekow (2005), it seems likely that in the same context, an attacker could scan an individual's shopping to reveal the drugs they are carrying, and loosely, identify them as a HIV patient. Thus, more context than the three properties considered in this model, are needed to take into account a fuller set of security issues.

On the other hand, domains which demand "high" security for RFID systems are generally those systems which identify a person. These systems include "implant based medical systems" and "electronic passport systems". In considering implant based systems, such as the VeriChip which is a human implantable tag; this tag contains usually only a serial number (Halamka et al. 2006). These tags also operate at 125Khz, making them relatively short range devices. Consequently, with the tag firmly embedded in a person, obtaining personal data would require access to the database which contains this data, in addition to the attacker getting extremely close to the individual. While attacks such as tag relaying could extend the range of such attacks, or existential cloning provide the serial number without physical intervention; this would put a system's security on par with other serial-number-only systems. Thus, when considering the derived classifications, this model appears to make classifications on the basis of limited information.

When considering the above approach to security analysis, it seems likely that many of the influential system properties are not considered when a classification is derived. This may mean that a system's security requirements may not be as effective if derived through this approach.

3.4 SUMMARY

The previous chapter suggested the examination of security in a whole system on the basis of attack scope and constraints in various parts of the system. It seemed likely that understanding the relationships in system security could lead to practical security improvements. To this end, this chapter set out to examine how previous work on RFID and security analysis approaches would facilitate this.

When considering the examples of various analysis approaches of security in RFID systems, it seems likely that much of the focus has been on the RFID technology. Work by Sarma et al. (2001) identified the influence of various 'synergistic effects' on the tag (for example: reducing cost by minimising the size of the integrated circuit, memory, logic and power circuitry) which affects security in RFID systems, but the focus was largely at the tag and reader level. Some authors have taken a more systematic approach to threat analysis such as Thompson et al. (2006), however, they did not consider these synergistic effects in systems, nor the interrelationships which occur throughout the system as security is introduced. These assessments and models have appeared to lack integration amongst system properties.

Conversely, Mitrokotsa et al. (2010) have provided a view of security in the context of system layers, which is on par with accepted views of the domain, but many of the attributes were fixed to attacks or solutions – something which should change in order to consistently reflect the variations which arise when consideration is given to security interrelationships in systems (for example: the changes to solution cost when addressing an attack at a layer different from which it occurs in). Moreover, Rotter (2008) illustrated that actual system security could be classified based on system properties, however, these classifications appeared to be limited to a few properties at the oversight of other influential system properties. When considering these examples, it seems likely that these would benefit if the capacity for capturing appropriate system information, especially context, is enabled - the derivation of security requirements which consider the 'whole system' may then be more achievable.

Thus, this chapter concludes with the thought that an alternative RFID security model should be proposed to make possible a 'whole of system' approach to the analysis of security.

Chapter 4

Methods for Reference Model Construction

4.1 INTRODUCTION

To enable a 'whole of system' approach to the analysis of security in RFID systems this chapter reviews methods which will be used to introduce an alternative security model, as well as methods specific to various security analysis goals. Recall that in Chapter 2, the need to consider more of the system in security solutions was apparent when the forms of cloning and the inherent constraints in RFID systems were reviewed. Following the review of previous work in Chapter 3, it was suggested that existing models would not facilitate this approach to security analysis. The methods examined here will offer an alternative analysis approach which can accommodate these requirements.

To this end, a method to define a systems architecture is reviewed as a basis for defining an alternative model. Methods thought suitable for the specific tasks of enumerating security information make up the second part of the review. These latter methods focus on three facets of security thought relevant to this problem domain: standard system operation, threats, and solutions. These methods are embodied within various paradigm areas. For example, domain modelling provides a general paradigm, while specific methods relevant to domain modelling such as object oriented analysis (OOA) and entity relationship diagrams (ERD) are reviewed within this theme. The reason is to convey the general sense of the 'themes' which facets of the reference model to be constructed will embody.

In the chapters which follow, an alternative model which embodies the 'whole of system' paradigm will be introduced and following chapters will illustrate how individual methods, when integrated into the model, enable the derivation of security information.

4.2 DEFINING A SYSTEM'S ARCHITECTURE

As this thesis aims to make possible a 'whole of system' approach via a model, seen as the most beneficial approach to take from the previous chapter, it reviews a method for constructing and evaluating system models.

4.2.1 REFERENCE MODELS

A *reference model* is constructed when a system's operations need to be more effectively understood. A reference model is essentially a generic *blueprint* of a system type which contains the desirable properties of systems. Working from a reference model should lead to higher quality outcomes for solutions derived from the model as it embodies the system's most relevant qualities. (Fettke and Loos 2003).

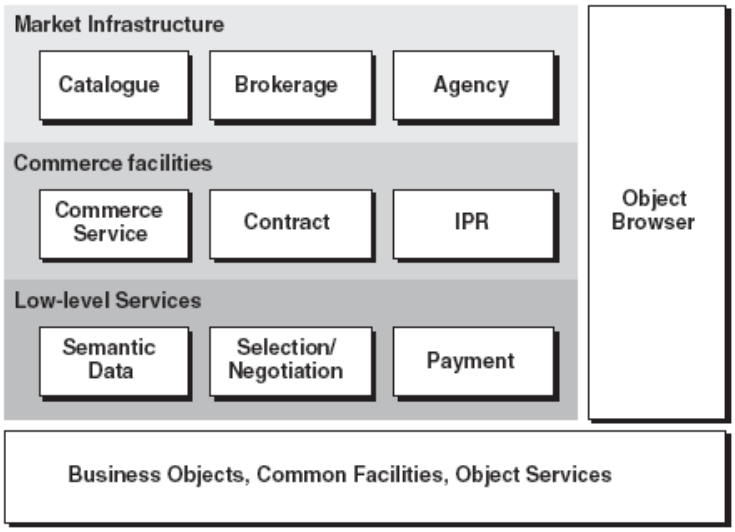

Figure 11 - Object Management Group (OMG) reference model for electronic commerce
A high level framework for specification of requirements for electronic commerce systems. (Mišic and Zhao 2000)

Other more tangential outcomes are possible using a reference model (Mišic and Zhao 2000). Some of these outcomes include: the model being used as a framework to standardise representations or communication amongst stakeholders; to develop more specialised models for specific scenarios; to map out specific system architectures; and finally, a widely accepted reference model leading to an architecture based development process. For security analysis in RFID, this may be beneficial, as Chapter 3 has suggested; security does not currently appear as wanted considering the influences of cloning and constraints. Thus, working from such a model may lead to more effective security analysis, and subsequently, more effective solutions.

A reference model facilitates increased understanding of a system by modelling a system's architectural properties - usually by examining individual system elements and modelling these elements along conceptually similar system functions enables derivation of these architectures. For example, one way of modelling elements in systems, is to hide individual characteristics using *layers*. Layers have found widespread appeal largely because of the Open Systems Interconnection (OSI) model (Zimmermann 1980) reference model (see Chapter 3). The OSI model, used in the communications domain, enables improved understanding of communication systems as it represents a communication system as a simpler system: having compressed all elements into a simpler structure, increased system understanding occurs as the focus is on what is contributed by each layer, rather than what each element contributes.

4.2.1.1 CONSTRUCTING REFERENCE MODELS

There does not appear to be an established method for constructing a reference model, and therefore, guidance on this matter has been sourced from the principles which are considered desirable for achieving high quality reference models. Misic and Zhao (2000) have proposed that consideration be given to a reference model's *syntactic*, *semantic*, and *pragmatic* properties in order for a model to achieve sufficient quality. These are now listed to form a basis for constructing a reference model:

- *Syntactic* properties describe how the model is represented in a modelling language. The Unified Modelling Language (UML) is a popular language for representing models, and one could consider a model's conformance to the rules of UML as a basis for possible model quality. (Mišic and Zhao 2000).

- *Semantic* properties describe how the model describes the domain; a model should be coherent in capturing meaningful elements which contribute to the overall goals of the model; and complete in capturing the amount of elements appropriate for the model's boundaries and scope, for it to be useable. (Mišic and Zhao 2000).

- Finally, *pragmatic* properties describe the association of the model with the intended audience. Pragmatic quality considers how well the model corresponds to the audience's interpretational needs. For example, system developers may find it desirable that a reference model focus on technology, whereas system analysts could find more benefit in a model of business rules. (Mišic and Zhao 2000).

When considering these properties, it seems likely that they are a suitable guide for ensuring a model exhibits quality, and they will be used when constructing the alternative security model.

4.2.1.2 COMPARING REFERENCE MODELS

While the above properties seem relevant for constructing and assessing individual reference models; the matter of evaluating reference models by comparison against each other is now considered. Fettke and Loos (2003) have suggested four reasons why such evaluation is necessary:

1. Evaluation leads to more effective understanding of each model's characteristics;
2. Similarities and differences in models can be identified;
3. Identification of which model are more relevant to a problem domain;
4. The strengths of each model's theoretical basis can be determined.

There a variety of ways in which reference models can be evaluated on the basis of quality. Fifteen approaches are summarised by Fettke and Loos (2003). These fall into various classes, ranging from *descriptive*, *theory based*, through to *empirical* methods. Some of these methods are awkward to use if reference models are even marginally different from one another. C*ase study* evaluation investigates a specific reference modelling situation in a particular application scenario. The quality assessment is derived from the outcome of the evaluated case study using the reference model. Conversely, *laboratory experiments* can evaluate the model in an environment which is unbiased, as the influence of independent variables on dependent variables can be controlled.

To summarise, reference models offer an accepted approach to modelling a system. The use of the approach, in developing a model of security in RFID systems, would be a distinct advantage as it offers a formal development process when quality criteria are considered, as well as offering accepted ways to evaluate model usefulness. This may lead to the development of a strong representational basis, which can enable the 'whole of system' approach to the analysis of security in RFID systems.

4.3 ANALYSING FACETS OF A SYSTEM

This section reviews methods which could be integrated into the reference model to achieve various analysis requirements. An apparent limitation on a reference model is that the architectures must be relatively stable – the fact that a system could be demarcated by layers may never change – however, the elements of the systems which are hidden by the reference model could change. As analysis moves from the reference model to actual systems, as the level of abstraction decreases, the similarities between systems at an element level may vary. Consequently, understanding the analysis methods which could be facilitated by the reference model would be useful in order to choose which would more effectively yield useful information when used for the analysis of a specific system.

From previous work reviewed in Chapter 3, it appears that three general concepts are relevant to the analysis of security in RFID systems:

- Analysis of actual RFID systems based on their functional capabilities under standard operation – without security matters influencing the behaviour of the system.

- Analysis of security threats which result in a system moving from a standard operating situation, to one where operations are invalidated by attacks.

- Finally, analysis of solutions in the context of system properties in conjunction with threats, to achieve practicable security requirements.

The basis for the inclusion of these facets in a reference model is established from the review of previous work on models in the previous chapter, Chapter 3. To this

end, methods which may allow for the derivation of information in the above areas are reviewed. In the coming chapters, the reference model will facilitate the integration of these methods for achieving specific analysis outcomes.

4.3.1 DOMAIN ANALYSIS

One method to define what broadly constitutes a system's standard operations is *domain analysis*. Domain analysis can refer to a variety of approaches:

- The process of creating reusable views – usually a domain model - of components in a domain for creating a software representation of a domain (Pressman 2000).
- The knowledge acquisition stage in the development of expert systems (Prieto-Diaz 1987).
- Conceptual modelling and knowledge engineering in addition to software engineering (Zand 1998).

A domain model forms the basis for understanding how system components function. To determine how a system functions, usually two analysis stages are undertaken: *data analysis* and *classification*. The *data analysis* stage is when a domain's basic elements are identified as entities, operations, events, or relationships. The *classification* stage uncovers *information structures* which characterise classes of elements. (Arango 1994).

The classification stage can give rise to a taxonomy of objects, which allows coverage of entity clusters via abstraction. The branches in the taxonomy become the relationships between the entities thereby communicating domain rules. From the taxonomy, a *controlled vocabulary* can emerge which enables identification of the domain concepts. Thus, the domain model can provide not only generic views of system elements, but also a set of terminologies, to assist one to understand the domain. (Arango 1994).

It is worth noting that domain analysis is an ongoing process, and thus, a domain model is generally never fully completed. Analysis is repeated a number of times depending on the amount of detail required (Zand 1998). Once reusable components

are identified and added into a domain's abstraction, the reuse data are gathered and fed back to the domain analysis process for tuning and updating of the domain model. The process of domain analysis repeats itself until a satisfactory level of accuracy has been reached.

As a basis for understanding how security is needed in RFID systems, it makes sense to begin by understanding what constitutes the elements of a system. The following review examines methods which can be applied in a domain model approach.

4.3.1.1 OBJECT ORIENTED ANALYSIS

As RFID contains a variety of objects, one may look to identify and model these in a logical view in order to capture system components.

To model the standard objects for a domain model, Object Oriented Analysis (OOA) represents the content and behaviour of domain objects. It results in various OOA diagrams, one of which is the *class diagram*. This diagram generalises objects into templates called *classes*. These diagrams can be depicted using the Unified Modelling Language (UML) (Bruegge and Dutoit 2004; Maciaszek and Liong 2005).

Some OOA concepts are now reviewed: *objects*, *messages*, *associations*, and *multiplicity*. These concepts, in particular, form the constituent parts of an OOA class diagram and can be represented in UML.

Objects are instances of classes. They are entities which can be created, modified, or destroyed during the life of a system. An object has a state that includes the values of its attributes and its links with other objects. Objects have attributes (such as shape, weight, colour, and type of material). Operations represent behaviours of an object, whenever an object receives some stimulus, called a message, it initiates behaviour. (Bruegge and Dutoit 2004; Maciaszek and Liong 2005)

Messages allow objects to communicate with other objects. A message stimulates behaviour to occur in the receiving object. An object requests the execution of an operation from another object by sending it a message. The message is matched up with a method which carries out the intentions of the message. (Bruegge and Dutoit 2004; Maciaszek and Liong 2005).

Associations are relationships between classes. Associations represent relationships between objects e.g. objects which compose other objects. Associations between classes are made up of *types* and *multiplicity*. Association *types* include a "has-a" association for depicting structural interactions between a system's components. Conversely, as a way of indicating the number of associations that can originate from an instance of a class, *multiplicity* is defined at the ends of each association. There are various types of multiplicity. For a one-to-one (1:1) association, exactly one link exists between instances of each class. A *one-to-many* association has a multiplicity one (1) on an end, and zero to many (0…n) on the other end. A *many-to-many* association has a multiplicity of zero to many (0…n) on both ends, thereby denoting that an arbitrary number of associations could exist between instances of the two classes. These are a means of depicting rules of interaction between components in a domain at the logical layer. (Bruegge and Dutoit 2004; Maciaszek and Liong 2005)

To this end, OOA can be used in conjunction with UML to model the objects in a domain in a formal way. For a 'whole of system' model of security in RFID systems, this type of approach may offer the advantage of standardising depictions of components such that security proponents could agree on what broadly constitutes the standard operations of RFID. Knowledge of the logical system would therefore provide a basis for understanding what constitutes invalidation of parts of the system using attacks at the logical layer.

4.3.1.2 ENTITY-RELATIONSHIP MODELLING

While OOA models the logical layer – static objects and the relationships between objects - in a system like RFID, where object interactions lead to the production of RFID data, this data can also be modelled to provide system understanding. As data is produced when a tag and reader have interacted, modelling the data would in effect be modelling associations which have taken place at the lower RFID system layers - for example, when tags and readers interacted via radio signals in physical environments. Adding these details to a model of RFID could contribute detail to the view of the whole system, thereby further enhancing system analysis capability.

To this end, *entity-relationship modelling* is briefly reviewed here. This method usually results in an Entity-Relationship Diagram (ERD) which represents relationships in data (Pressman 2000). It separates the data from the class or object

which has been defined in the Object Oriented Analysis (OOA) – thus, it defines a data view. Originally credited to Chen (1976), his approach adopted the view that the real world consists of entities and relationships which are characterised by semantic information. Only a brief description is given here, and a comprehensive overview can be found in Benyon (1997).

Many of the concepts found in OOA are to be found within ERD but there are some differences as well (Bagui and Earp 2003). In ERD, the *object* is referred to as the *entity*, although the entity is similar to an object, having attributes and associations. *Relationships* provide a way for associations to be modelled at the data level. Teorey et al. (2006) discussed several semantic properties to capture enterprise rules using relationships: *degree* and *multiplicity*. *Degree* specifies the number of entities participating in a relationship: a *binary* relationship has degree two (two entities); a *ternary* relationship has degree three; and to be non-specific, n entities participate in an *n-ary* relationship. *Multiplicity* specifies the number of instances of each entity in a relationship. The basic connections are: one-to-one, one-to-many, and many-to-many. These various relationships are focused on *events* which occur between entities. These differences are the way ERD can focus on modelling data when entities or objects have interacted in parts of a system.

4.3.1.3 FEATURE CONSTRUCTION

Through object interaction in a system, *data* can emerge. In RFID systems, this happens when tags are read by readers (Garfinkel and Rosenberg 2005). The data which is produced is a combination of: a tag serial number, a reader serial number, and the timestamp of when the interaction took place. While OOA could model the components in an RFID system – tags and readers – and ERD the relationships at the data layer – the *context* surrounding these interactions is, at first glance, relatively sparse.

When considering only the above RFID data features, it seems likely that one could deduce very little information about the system. Consequently, on the basis that few data features are established through such interactions, in establishing a comprehensive view of an RFID system, it may be necessary to define a substantial amount of context to understand what exactly the RFID data means. However, this has the disadvantage of making a representational model highly application specific.

Thus, an alternative to be explored in this section is how features can be produced without having to provide application context.

Feature construction is a method which enables the derivation of features from existing features in data (Alfred 2008). The approach is used when more information needs to be gained from sparse data. This is different from *feature extraction* which looks for classes which exist amongst values in a dataset and uses these to derive additional features as a way of classifying records which share those values. Although feature construction is associated with the machine learning domain it also has relevance outside of that domain.

Several types of approach exist (Alfred 2008), and these are briefly reviewed. *Hypothesis driven* methods construct new features based on a previously generated hypothesis or discovered rules (Alfred 2008). A hypothesis is constructed and examined to construct new features. These new features are then added to the set of original features to construct another new hypothesis. The process is repeated until a stopping condition is satisfied, although this is highly dependent on the quality of the previously generated hypothesis. *Data driven* methods construct new features by directly detecting relationships in data (Alfred 2008). However, other approaches appear less reliant on an automated approach, and thus suitable for modelling systems manually. Of these methods, *knowledge driven* methods (Wnek and Michalski 1994) – are the most common approach (Tan et al. 2006) to feature construction and appear to apply expert domain knowledge to construct features.

Using feature construction in the RFID domain may allow for alternative features to be listed, which are generic to a class of RFID systems. This may further enhance a model of standard system operations, providing a context around sparse RFID data.

To summarise, methods have been reviewed which could be used to derive parts of a domain model. A domain model would provide a logical and data basis for understanding how RFID systems are supposed to function, and therefore, a way of identifying when they are not functioning as they are supposed to. While this section has encompassed a variety of methods, both overarching and specific, the following chapters will show how a solid domain model provides a good basis for approaching security analysis on a 'whole of system' basis.

4.3.2 THREAT ANALYSIS

As the focus of this thesis is the security of RFID systems, methods which can analyse the threats to systems should be reviewed. Approaching this systematically would offer the advantage of identifying how a system influences attack instantiation, as well as solution implementation. A method which makes this possible is the attack tree method and this is now briefly reviewed.

4.3.2.1 ATTACK TREES

Attack Trees are a threat modelling method proposed by Schneier (1999, 2004). They are an *attacker centric* approach to deriving a depiction of the ways a system's goals can be invalidated. Attacks are modelled from an attacker's perspective as a *tree structure*. The root node of the tree represents the attacker's attack goal and the leaf nodes depict ways of achieving that goal. As an attack can be a goal, it can be further decomposed into further attacks – in effect, this can be a recursive process until sufficient decomposition has been attained.

Attack trees can be augmented with logical operations and by assigning values to attack tree nodes. Two logical operations can be used for this purpose. Conjunction (logical "And") between nodes, represented as a diagrammatic arc, indicates a dependency between nodes in achieving a parent goal. Disjunction (logical "Or") between nodes is the default state, and does not have any special diagrammatic symbol. It specifies that there are different ways of achieving a parent goal. Values can be added to each node to signify, for example, the cost and skill required, or whether an attack is legal. Adding these constructs to nodes is intended to enhance an attack tree's semantics; however, they have the distinct disadvantage of introducing specific application detail making them application specific. (Schneier 1999, 2004).

When these constructs are specified throughout the branches of an attack tree, it is then possible to determine the likely sequences an attacker could choose. The node values in a sequence are aggregated to derive an indication of the overall value of the sequence. For example, the *cost* of an attack sequence could be derived by adding together the cost of each component. When such calculations are applied to all branches, the branch of least cost could be identified, which could be the most likely sequence the attacker chooses, for example. However, as these values are

instantiated for a specific system implementation, again, these make any particular attack tree specific to that system. (Schneier 1999, 2004).

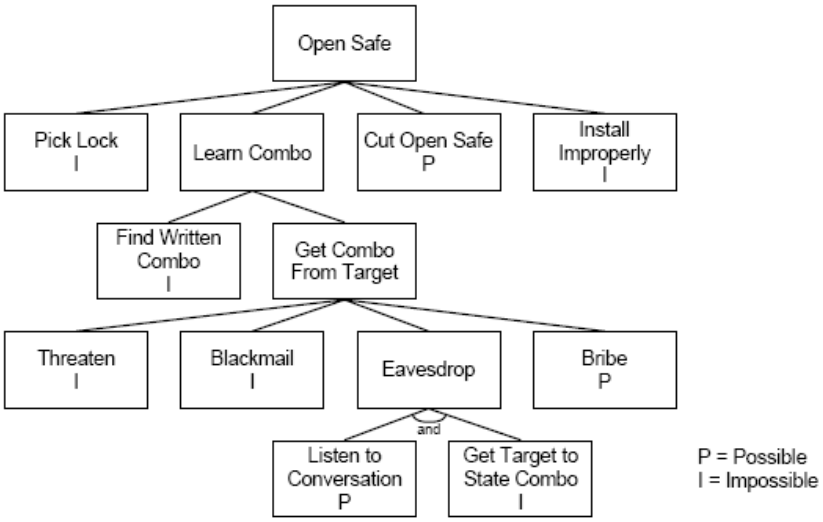

Figure 12 - Attack tree for opening a safe
This attack tree represents the series of attacks an attacker may enact to obtain access to a safe. (Schneier 1999)

To indicate an attacker's behaviour, the nodes of an attack tree are traversed from leaf node to root node, or vice versa. As each node is the composition or decomposition of an attack, the steps in achieving the attack goal are a sequence of attacks. As illustrated in Figure 12, for example, in order for an attacker to open a safe, the attacker could learn the combination by getting the combination from the target, and this could by achieved by eavesdropping on a conversation where the combination is being discussed (Schneier 1999). Each sequence through an attack tree represents a way the attacker behaves to attain the attack goal. This suggests that a systematic approach to analysing threats can be achieved using this method. (Schneier 1999, 2004).

Attack trees have been used previously in RFID research for modelling threats against privacy. The ways tags reveal information about the objects they are associated with has been analysed by Spiekermann and Ziekow (2005) using this method. However, recall from Chapter 3 that the lack of alignment to system layers was a perceived shortcoming as the method lacked the ability to determine which security solutions were feasible. This shows that attack trees are an accepted way of

modelling RFID threats, but that there are outstanding problems to be solved in their use in RFID security.

Attack trees are not the only threat modelling approach which would be useful for analysing attacks in an RFID system. The main benefit they offer is the ability to structure attacks hierarchically – which can be linked to the architectural property of RFID systems (see Chapter 3).

4.3.3 SOLUTION ANALYSIS

One approach to analysing solutions on a 'whole of system' basis is to study solutions in the context of actual systems. However, some authors report that this is not so straightforward for researchers. In our work on developing an intrusion detection system for RFID, Mirowski and Hartnett (2007) reported that the availability of actual systems for examining security was limited. We had to evaluate the intrusion detection system using, not an actual RFID system, but sanitised RFID data injected with synthetic attacks from a system. This had the disadvantage of being an artificial representation of what attacks may look like rather than being an actual representation. As the data was sourced from a live system, any attacks which may have been prevalent but unbeknownst to us may have influenced the accuracy of the results. In addition, very little information was available about the users of the system, and thus, the context of the data was minimal.

Recognising that the availability of RFID data was also an ongoing problem for other researchers, Mirowski et al. (2008) released RFID data on the internet for other researchers to use. Since releasing the (presumed) attack free output data on the internet, it has had over 900 downloads[1]. This suggests that there is an ongoing need for actual systems to be available for analysis of solutions. To this end, this section reviews a method which may be suitable for analysing solutions for security in RFID systems on a 'whole of system' basis.

4.3.3.1 AGENT BASED MODELLING AND SIMULATION (ABMS)

ABMS is a simulation methodology which models a system as a collection of agents and the relationships between those agents. Although what exactly constitutes an

[1] Statistics were viewed on 16/12/2010 from the University of Tasmania electronic prints database at http://eprints.utas.edu.au/es/index.php?action=show_detail_eprint;id=6903;

agent is contentious, in general, an agent executes various simple independent behaviours (Macal and North 2005; Korth 2006). While the individual rules of each agent could be simple, the model's agents collectively exhibit more complex behaviours than a single agent. This is called emergent behaviour. This is seen as beneficial, as often systems are easier to understand as constituent components (Bonabeau 2002). Thus, highly complex systems can be modelled using relatively simple components, whilst still attaining the behaviour of the 'whole system'. An example of this emergent behaviour is that of termites working together to create large mounds that have very complicated temperature control structures. Even though no single termite plans to produce a specific mound, the mound emerges through termite interactions. In simulations of similar phenomenon, the design of each agent is simple, but the whole system emerges through these simple interactions.

From a review of general simulation literature (Robinson 2004), it seems likely that consideration should be given to two general simulation issues: firstly, how the simulation can be implemented – and thus, how to implement the ABMS; and secondly, the methodological approach which is to be taken to simulation development. These are briefly reviewed and related back to the concept of ABMS.

ABMS can be implemented more easily using toolkits (Gilbert and Bankes 2002). An example toolkit is the Multi-Agent Simulator of Neighbourhoods (MASON) (Balan et al. 2003; Luke et al. 2004). MASON provides some of the core elements needed for ABMS: *modelling* and *visualisation*. It allows a modeller to define agents as entities. These can be scheduled to perform some action inside a continuous *virtual* environment. Visualisation can occur in a three-dimensional viewport which animates agent interactions. MASON is an extensible toolkit allowing a modeller to make customised simulations. For these reasons, MASON has been used for a wide range of multi-agent simulations, ranging from swarm robotics to social complexity.

The relationship of developing a simulation to using a methodology is now considered. The procedures to develop a simulator, have been discussed in Robinson (2004) and Law (2005). Robinson (2004) has proposed that simulators have *modes*; ranging from highly accurate representations of systems for predicting outcomes in real systems (Mode One), to less formal representations, which facilitate a group of

individuals through discussions which take place during the modelling process (Mode Three). These modes give direction to the approach one may take in developing a simulation for the purpose of system analysis.

For RFID simulation, one assumes that there would usually be a single security proponent involved in an analysis task. Thus, for a 'whole of system' approach, *Mode Two* simulators appear to be the most relevant method.

A Mode Two simulator is developed for problem understanding and problem solving by a single modeller. These are seen as a process of 'social change' as learning occurs through the process of development, as well as through experimentation with the simulator. Model users are highly involved during the modelling process, gaining benefits from all stages in terms of an improved understanding as well as the solutions that could be derived from experimentation with the model. These users are the direct beneficiaries of the modelling process. Consequently, validation is considered in terms of whether the model is sufficiently accurate for its purpose and is performed by the modeller in conjunction with the users.

For a 'whole of system' approach to RFID security, the main benefit ABMS would offer is the ability to simplify systems into their constituent components. This could allow an analyst to examine tags individually, or the 'whole system', depending on analysis goals. In implementing an ABMS, one would be providing a mechanism for the analysis of solutions prior to actual system investigation.

4.4 SUMMARY

This chapter has reviewed existing methods that will be used in the coming chapters to derive an alternative security model which makes possible a 'whole of system' approach to security analysis.

The reference model method will be used to derive an alternative representation of security in RFID systems to facilitate the 'whole of system' approach. This method offers an overarching approach to defining a system's architecture. The principle of quality ensures that a suitable representation would be derived. Another benefit is the means of evaluation to compare a derived model to existing work. Using this approach, a more robust model of RFID could be derived when compared to existing

models, and thus, such a model may be more suitable to facilitating the 'whole of system' approach.

Moreover, a variety of methods which will be applied to a reference model were reviewed for the specific purpose of identifying how these would contribute systems information. These came from a variety of domains but if integrated in a reference model, would be made to work for the specific goal of analysing security on a 'whole of system' basis. It is important to make this requirement clear, as without a representational basis, the results which would be derived from each analysis method may not be integrated across the whole RFID system nor related to one another. In using these methods this concept of integration is a perceived necessity as what is proposed is that threats and solutions are considered in relation to a domain context.

This chapter ends with the thought that a 'whole of system' approach can be facilitated by a reference model, and individual methods, which have systematic qualities, can then be applied to the model, in order to achieve specific security analysis information. Consequently, the next chapter introduces an alternative model for the 'whole of system' approach and successive chapters illustrate how 'whole of system' analysis achieves specific analysis outcomes by integrating the methods reviewed above in the model.

Chapter 5

An Integrated Layered and Partitioned Reference Model

5.1 INTRODUCTION

In Chapter 3, in order to determine how facilitative existing approaches are to a 'whole of system' approach to RFID security, previous work on security analysis for such systems was reviewed. Models imparted the most concise representation of security matters, and two models in particular, illustrated the benefits of examining security in relation to RFID systems but their limitations were apparent. Rotter (2008) proposed a model which enables system classification, however, his model does not appear to be suitable for analysis beyond several system properties. Conversely, Mitrokotsa et al. (2010) proposed a model which enables analysis over system layers, however, the use of security principles and attributes only within individual layers reduces the model's generality. In summary, it seems that previous work has been localised to specific system properties, which has the drawback of missing the interrelationships which are relevant throughout the 'whole system'.

To address these apparent limitations, this chapter introduces and describes an alternative model which makes possible a 'whole of system' approach to the analysis of security in RFID systems. It is distinguished from previous models which have been reviewed in Chapter 3, on the basis of integrating layer and partition properties, and is therefore entitled, *An Integrated Layered and Partitioned Reference Model.* The layers are: real world, RFID, and strategic. Conversely, the partitions are: standard operating, problem, and solution. The model integrates these layers and partitions using a reference model approach. The architecture of this model is explained and how this will facilitate a 'whole of system' approach is described. Successive chapters will expand on the use of the approach, by integrating individual methods through the model's structure for specific analysis goals and present evidence to support its advantages over previous work.

5.2 THE PROPOSED REFERENCE MODEL

The alternative model, illustrated in Figure 13, is based on the reference model paradigm (Fettke and Loos 2003) and is distinguished from previous work reviewed in Chapter 3, by integrating *layer* and *partition* properties. Layer properties are horizontal in the model to capture system elements. Conversely, partition properties are vertical in the model to capture security elements. The model is non-prescriptive

as to what constitutes the elements of the system; leaving these details to the security proponent in their use of various modelling methods, however those should be modelled around the definition of a particular layer or partition.

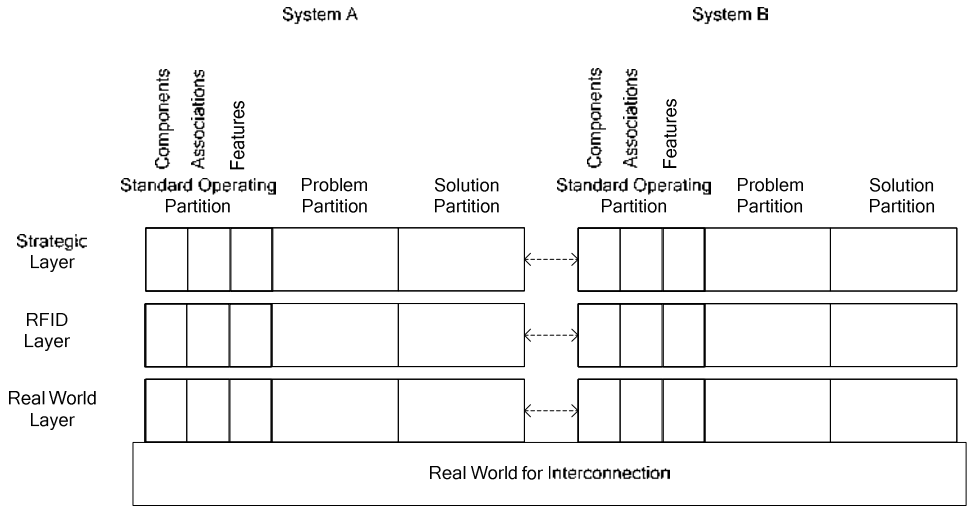

Figure 13 - The integrated layered and partitioned reference model

The rest of this chapter explains the model's structure, on a per property basis, and will conclude by describing how these properties are integrated to facilitate a 'whole of system' approach to the analysis of security in RFID systems.

5.2.1 LAYERS

The *layer* property of the reference model is illustrated separately from partitions in Figure 14. A horizontal layer captures a composition of system elements along functionally similar lines. In the reference model, systems are constituted not only by the RFID technology such as tags and readers, but also the physical world and the strategic layer which defines the system owner's information goals. To attain this broader perspective of what constitutes an RFID system, the reference model uses major abstraction layers. In contrast, previous work (Avoine and Oechslin 2005; Shepard 2005; Mitrokotsa et al. 2008, 2009) has usually focused on specific system layers mostly concerning RFID technology, which has the drawback of excluding the interplay between the environment and enterprise.

To this end, three major layers are introduced in this model: real world, RFID, and strategic. The *real world* layer captures the application environment in which an

RFID system is situated, specific to that system. The *RFID* layer captures the RFID technology such as tags, readers, and middleware. Finally, the *strategic* layer captures the company's information goals in using RFID in the particular application environment. This model takes a broad view, using these major horizontal abstraction layers, of what constitutes the 'whole system', when compared to previous work in Chapter 3.

The *real world for interconnection* grounds systems which are under comparison. It represents the environment which is common to these systems. For example, in a pharmaceutical supply chain, it may be the 'supply chain' (which contains several RFID systems each located at different distribution centres). It is modelled separately from the real world layer as it represents the common environment and not the local environment of each RFID system (the local environment is represented in the real world layer). If a single system is modelled, then the use of the 'real world for interconnection' is irrelevant.

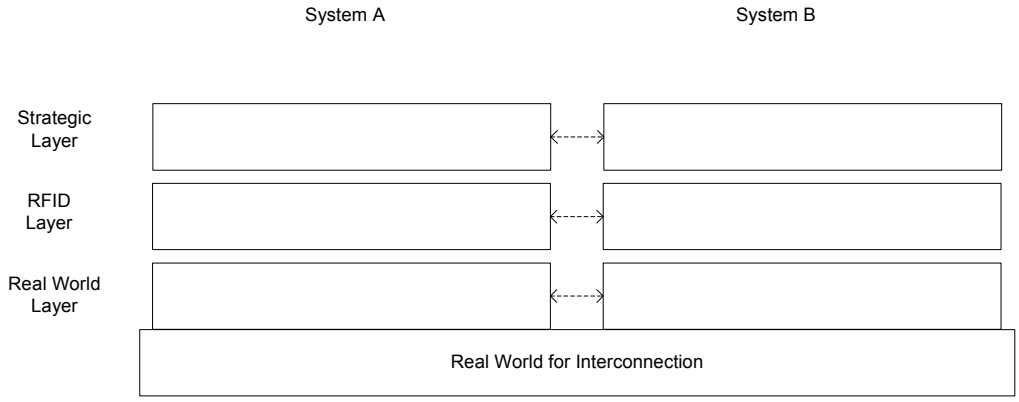

Figure 14 – The reference model depicting the major layers
These are represented as horizontal separations around which minor layers can be decomposed to organise system elements.

Within each of these major layers, the principle expounded in this model is that additional layers can be included. To attain greater detail of the RFID layer, for example, any of the previous work on modelling RFID as layers could be used – provided they are layered. These would include previous work on models based on Open Systems Interconnection (OSI) layers (Avoine and Oechslin 2005; Shepard 2005) at the RFID layer. As this model takes a broader view of what constitutes a system, it remains extensible to capture existing work on RFID systems modelling.

This is a feature which is not apparent in any previous work that was reviewed in Chapter 3.

This model achieves *hierarchy* of layers, a key characteristic of RFID systems, as seen in previous work on RFID models (Garfinkel and Holtzman 2005; Ranasinghe and Cole 2008), by ensuring a strict ordering between the major layers. It is assumed in this thesis, that the lowest layer is the real world layer, as all systems are going to have some sort of physical environment in which they operate. Next, the RFID layer is modelled, as it would be implemented into a particular application environment. Finally, the strategic layer is modelled, as information goals can only be achieved on the basis of an RFID layer having been implemented into an application environment. The concept of hierarchy via layers is therefore apparent in this model and is further explored in the next sections.

5.2.1.1 REAL WORLD FOR INTERCONNECTION

While not a layer which contains individual system elements, comparisons between RFID systems can be performed in this model on the basis of the *real world for interconnection*. It is a fundamental concept which represents a common environment to all systems which are under an analysis. An early representational concept for showing *interconnection* between systems can be found in the OSI model specification (Zimmermann 1980), and on the same basis, of representing interconnection between communication systems, this is reused in this model. To illustrate this ability, two sets of layers are depicted: *System A* and *System B*. However, if a single RFID system was being analysed then only one set of layers, and hence system, need be used, as no comparison is being made.

The ability to compare several RFID systems on the basis of common layers is a feature which is distinct in this model, due to the fact that this layer is included as distinct from the real world layer. This enables the making of general comparisons between systems. Rotter (2008) has attempted to achieve this comparison in his model using various system properties such as 'openness'. However, not every system can be evaluated on the basis of only a few criteria. As the only commonality amongst systems in the proposed model here is the system layers, the advantage imparted is that systems can be compared across any layer, and hence, any system property can be utilised during analysis.

5.2.1.2 REAL WORLD LAYER

The *real world layer* contains a system's local application environment and its characteristics. Local application environments can include: supply chain 'custodians', 'zones' in a toll way, or 'rooms' in a building. Depending on the level of abstraction needed, characteristics could be modelled using geometric concepts like those defined by a coordinate system such as the global positioning system (GPS). Other elements in the local application could also be modelled and may include: physical entities, movement constraints, or rules of interaction between objects. The advantage is that consideration can be given to the influences the local environment has on the RFID system.

It is worth noting that the real world layer is distinguished from the physical layer which has appeared in previous work (Mitrokotsa et al. 2008, 2009). Previous work referred to a physical instantiation of RFID components, whereas the real world layer in the proposed model captures the physical environment and components relevant to the system but which are not RFID components (for example: physical entities). This allows the model to capture more of the constraints imposed by the physical environment when compared to that of previous work (Mitrokotsa et al. 2008, 2009). This is an important feature in an RFID security model, as often RFID implementations are largely governed by physical constraints in the real world. For RFID security, this could be the factor determining whether a system is designed properly, and potentially more secure, or poorly designed and thus potentially less secure.

5.2.1.3 RFID LAYER

The *RFID layer* captures the RFID technology which broadly constitutes what previous work has always focused on as constituting the *RFID system* – recall this heavy RFID focus from Chapter 3. In the proposed model, tags and readers operate at this layer through radio frequency and anti-collision protocols, as does the middleware which links the RFID technology to the rest of a company's information systems. As this model takes a broader view of what constitutes an RFID system, RFID technology is contained in a single layer, which downplays the influence of the technology on analysis. What is provided here is a balance between the RFID technology and the physical and strategic layers.

The representation of the RFID layer as the centre layer of the proposed model is intentional. It conveys an overarching concept generally expounded by RFID - that RFID *bridges* the physical world to the digital world of a company. In previous work, for example Mitrokotsa et al. (2008, 2009), this central role that RFID plays has not been depicted as centralised, to some extent downplaying the centrality of the RFID system between the environment and enterprise. Conversely, in this model, as the RFID layer is the central layer, it suggests that RFID security should be considered central to the analysis task, and yet at the same time, RFID should be considered in the context of applications and the information goals of the company. Thus, this addresses the perception that previous work has appeared to focus more on RFID technology (see Chapter 3).

As the reference model has been derived using layered decomposition, more layers can be used to attain granularity. For the RFID layer, this is where the use of previous work on RFID layers (Avoine and Oechslin 2005; Shepard 2005; Mitrokotsa et al. 2008) could be introduced. Also, representations from previous work that have taken less formal approaches to layers (Hassan and Chatterjee 2006), could be included. This can be achieved not by decomposing the major layers into finer layers, rather, simply inserting the less structured representations, such as taxonomies, inside the major layer.

Centrality of this layer in the model and its intended further decomposition into minor layers, addresses limitations which are apparent in previous work, whilst ensuring that it remains specific to modelling RFID systems. For these reasons, the replacement of the RFID layer for a more generic ICT layer is not expected and its possible general use is beyond the scope of this thesis.

5.2.1.4 STRATEGIC LAYER

The *strategic layer* captures the information goals of the company which implements an RFID system. If a general approach to analysis is taken, then this layer can capture the generic information goals of RFID.

The concept of a strategic layer has been obtained from Mitrokotsa et al. (2008, 2009). In their model, the strategic layer captured various system properties, such as: costs vs. utility tradeoffs, logistical factors, and real world constraints. In this

alternative model, however, the concept of a strategic layer focuses on the information goals of the system's owner e.g. the company.

In doing so, through the introduction of a real world layer, the model further differentiates itself from the work of Mitrokotsa et al. (2008, 2009) as it leaves the logistical factors or real world constraints to the real world layer. It also leaves concepts like cost and utility tradeoffs to the discretion of the specific application which is being analysed through the model, choosing not to put these into the model. These values would be instantiated from specific analysis techniques. Consequently, it is therefore differentiated from Mitrokotsa et al. (2010) who chose to include these values in the model. The benefit of the approach expounded in the proposed model is that these values can be left to vary amongst actual system implementations, ensuring the model remains generic. These are derived when specific analysis methods are integrated into the model.

To this end, the example information goals for the strategic layer, suggested in this thesis, are the two proposed by Hassan and Chatterjee (2006): authorisation and monitoring. They have proposed information as sourced from RFID systems, to be used for *monitoring* or *authorisation* purposes. Both goals can be attained from the RFID layer with no change to the underlying system. Consequently, the decoupling of the strategic layer from the RFID layer, in this model, ensures appropriate de-emphasis on the reliance on the RFID layer in achieving strategic layer goals. This has an advantage over the model proposed by Rotter (2008), as the information goals in his model are largely fixed to the two criteria used to classify systems, whereas the proposed model can define any information goals a proponent chooses.

To summarise, this model uses horizontal system *layers* to capture system properties. Advantages are achieved over previous work as only three major, but relatively abstract, layers are presented – meaning less detail is fixed in system representation. To attain more detail, these layers can be decomposed using previous layered decompositions e.g. the OSI model at the RFID layer. Thus, the model is relatively extensible. The alignment of two system types enables comparisons to be made between systems on the basis of any layer due to the *real world layer for interconnection*. Finally, placing the RFID layer as the central layer indicates the

'whole of system' approach, across layers, is a balance between RFID, the real world and strategic information goals.

5.2.2 PARTITIONS

Vertical *partition* properties, depicted in Figure 15, capture the security concepts which have featured in previous work which was reviewed in Chapter 3: standard operations, threats (problems), and solutions. While not explicitly a structure which has appeared in previous RFID security models, partitions in this model are used to demarcate across the system layers, the threats and solutions which are associated with standard system operations. Thus, the paradigm expounded in this model is that in order to consider a threat, or solution, one must also consider what is standard for a system.

One could perceive the *columns* used by Mitrokotsa et al. (2010) to be *partitions*, however, these are not partitions in the same sense, as their model's columns do not represent distinct independent concepts that span all system layers. That is *potential damage*, *attack cost*, *class of tag*, *solution-cost* – all columns used in their model, depend on the *attack* column. Conversely, in the proposed model, partitions are complete security concepts, distinct from attributes which pertain to parts of a system or parts of security. This has the advantage of indicating that security concepts should be considered across all system layers.

In the proposed model, a partition divides the RFID domain using concepts relevant to security in RFID systems. To this end, the *standard operating* partition captures systems operating as intended by a company. The *problem* partition captures threats against a system. Finally, the *solution* partition captures any form of analysis which attempts to address threats in the context of systems. The use of partitions is a novel feature to RFID security representational approaches, and it will be illustrated that the primary advantage they impart is the ability to analyse security in a manner more linked to a system's properties.

It is worth noting that in this section, partitions are explained without reference to the horizontal system *layer* property – except for the inclusion of the *real world for interconnection* - to make clear the difference between layer and partition concepts. The next section will discuss how properties have been integrated.

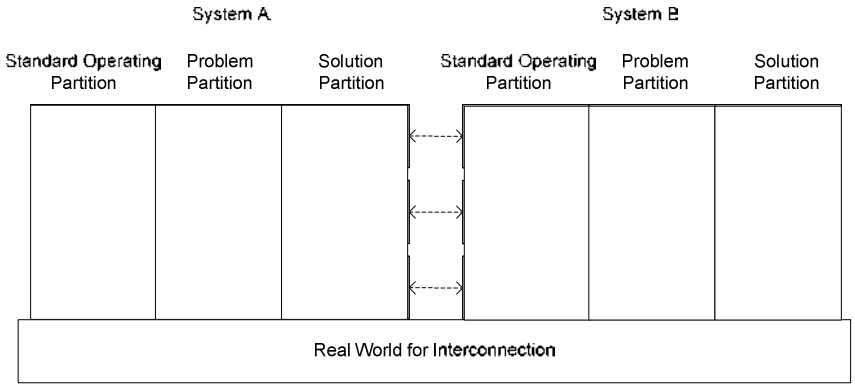

Figure 15 – The reference model depicting the major partitions
Partitions are modelled as vertical separators which span the full length of the layers enabling security concepts to be considered across the whole system.

As partitions have not appeared to such extent in previous work on RFID security models, the theoretical basis for their inclusion is briefly explained. Chapter 3 showed that it was often the case that systems in a normal mode of operation were discussed along with the threats and solutions which modify these standard operations. Partitions are a way of aligning these *concepts* in a single model. As they all pertain to the RFID domain, the RFID domain is demarcated from end to end by each security concept, as security is relevant to all ends of an RFID system.

The concept of layers is grounded in the *real world for interconnection* as a way of comparing systems. Partitions are a way of determining whether systems are *compatible* with security concepts. Having modelled the partitions of various systems and at each layer, comparisons can be made on the basis of each system's constituent parts. For example, taking several custodians' RFID systems in a supply chain, which share a common interconnection, one could analyse their standard operating partition to establish whether differences in operation exist, or alternatively, whether a solution in one system, would work in another system which was connected to the first system. Systems can be compared on the basis of what is constituted in each of their partitions and across each layer in a partition. Thus, a benefit of partitions is that comparisons can be made between system concepts to establish similarity of security.

This feature of the proposed model has not appeared to this level of detail when compared to previous work. Rotter (2008) has appeared to suggest that all systems of a particular system class are the same, however, they clearly differ along element

configurations resulting in different security classifications. The opposite was represented by Mitrokotsa et al. (2010), as they did not discriminate which solutions are feasible for particular application domains, appearing to suggest that a generic applicability exists. Consequently, the advantage imparted by the proposed model is that it can compare general system properties as well as specific systems.

Having provided a brief overview of the theoretical underpinnings of the *partition* property, each of the major partitions is now explained.

5.2.2.1 STANDARD OPERATING PARTITION

The *standard operating* partition captures the intended operating principles of RFID systems. This is where the standard domain components are modelled, such as in a domain model. As such it would be possible to subsume previous representations of RFID components in this part of the model. For example, the model proposed by Hassan and Chatterjee (2006), which has depicted different RFID components, could be inserted here as this partition is concerned with valid operations. Although their model depicts a taxonomic representation, the overlaying of these components over layers is not being considered in this section. This is discussed during the integration property later in this chapter.

5.2.2.2 PROBLEM PARTITION

The p*roblem partition* captures the security problems which affect RFID components which have featured in the standard operating partition. These are usually the threats such as those threats reviewed in Chapter 2.

Like the previous partition, as a relatively high level of abstraction of partitions is used, this partition can subsume previous work on threats. In Chapter 3, Thompson et al. (2006) illustrated that threats can be listed using threat analysis methods. This partition is where such a threat analysis would take place and the results situated. The improvement imparted by this model, over previous work, in particular the work by Rotter (2008) and Mitrokotsa et al. (2010), is that threat analysis if done in this partition, is automatically aligned with the standard operating partition and the solution partition, as partitions are aligned with each other. Thus, the context of actual systems is taken into account during threat analysis.

5.2.2.3 SOLUTION PARTITION

The *solution* partition captures the security solutions which could be implemented in the RFID domain. This is where all solutions for RFID security could be analysed, or alternatively specific solutions to different threats in the neighbouring partition could be aligned. Later in this thesis, 'whole of system' analysis towards solutions is illustrated using simulation, suggesting solution analysis can occur over system layers, and be aligned to threats and standard operations – a principle expounded in this thesis.

Conversely, Chapter 3 showed that previous work has not integrated the depiction of the relationships between solutions, the threats, and the standard operations of systems. This was a major shortcoming apparent in many previous attempts at modelling RFID security. However, this is something this model achieves *natively*. The advantage of depicting the solution partition last is that it imparts the suggestion that all solutions should be grounded in these earlier partitions. That is, one must give consideration to the solutions in relation to threats and the system in which threats and solutions are to be situated.

This is a point of difference over previous work by Mitrokotsa et al. (2010) that appeared to suggest the direct coupling of solutions to threats as the primary approach to take. The improvement is that direct coupling is only suggested as feasible when a basis exists in the standard operating partition. For example, if encryption is recommended in the solution partition, then there should be an associated threat in the problem partition, and moreover, the standard operating partition should have the necessary components which are under attack, but may also support the solution in some way at the tag layer. Thus, this model enables not only the encapsulation of generic security information, but also the modelling of specific systems for the derivation of specific system security information.

The use of partitions enables system and security interrelationships to be considered across concepts (domain, problem, solution) during the analysis of security in RFID systems – a requirement which became apparent in Chapter 2.

5.2.2.4 MINOR PARTITIONS

Layers can be decomposed to attain greater detail. Similarly, partitions can be decomposed into minor partitions. As partitions are an additional concept for representational approaches to RFID security, this section illustrates several minor partitions to show how partition decomposition can proceed

Figure 16 depicts the *standard operating partition* with the inclusion of three minor partitions. These have organised the standard components to consider: the components, their associations with each other, and the information which can be gained through their interactions, as separate concepts. These were derived using the various analysis methods discussed in Chapter 4, however, this section of the thesis backtracks slightly in order for the later chapters to make sense. Three minor partitions are introduced: *components*, *associations*, and *features*. The elements of these minor partitions will be further explored in the next chapter.

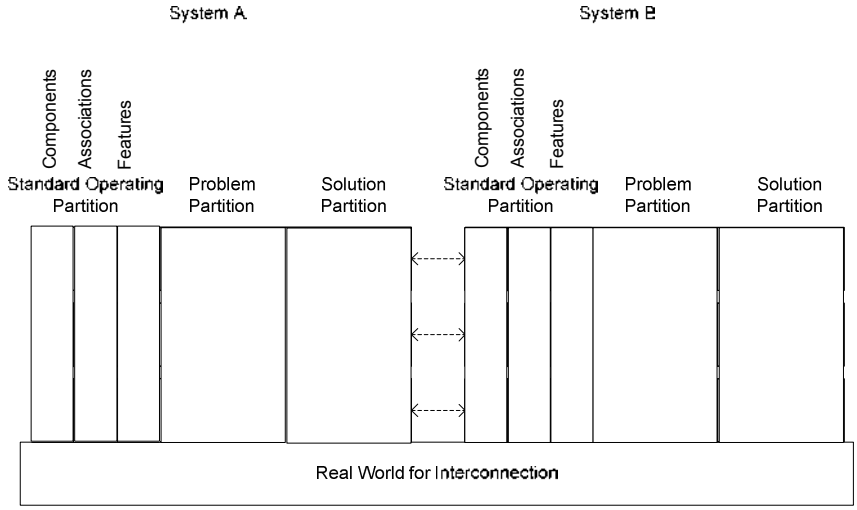

Figure 16 - The reference model showing minor partitions
These minor partitions are vertical decompositions of the major partitions as a way of achieving more security detail.

The *components* partition comprises the *logical* view of the RFID domain such as: physical entities; tags; readers; middleware; and a system's information goals. It focuses on the representation of discrete objects which would have attributes and operations. For example, in a pharmaceutical supply chain, in the real world layer, a

component would be a drug entity, whereas, at the RFID layer, a component would be a tag or reader. Thus, each object would have a logical template from which it is instantiated in this minor partition. In the next chapter Object Oriented Analysis (OOA) and the Unified Modelling Language (UML) (Bruegge and Dutoit 2004; Maciaszek and Liong 2005) are used to represent the components of this minor partition.

The *associations* partition comprises the *data* view which emerges when components in the component partition interact. These associations could be permanent associations if objects are associated with each other for the duration of their lifetime – such as the assembly of ingredients of a drug entity – or they could be temporary associations – such as the association which forms when a physical entity introduces a tag entity to a reader entity. In the next chapter, a formalisation of these *associations* for the RFID data layer will be introduced using Entity-Relationship Diagram (ERD) methods (Pressman 2000).

Finally, the *feature* partition represents metrics which can be constructed from the data associations between components in the previous two partitions. While there could be observable features at any of the layers, the next chapter will demonstrate the advantages which are imparted when features are constructed at the data layer. Manual knowledge driven feature construction approaches (Wnek and Michalski 1994) will be used to derive new features without the need to add additional systems context, illustrating how this partition can be analysed.

In this way the integration of minor partitions serves to enhance the granularity of analysis.

5.2.2.5 ABSTRACTION PRINCIPLES OF THE MODEL IN MINOR PARTITION ORDERING

The decomposition of minor partitions follows a simple principle that minor partitions should be abstracted such that the left-most minor partition is the most concrete, and the right-most partition is the most abstract. For example, for the standard operating partition: the *association* partition emerges from the *components* partition, and the *feature* partition emerges from the association partition.

In this way, what is analysed occurs across increasing levels of abstraction – from the most concrete concepts (the extreme left) to the most abstract concepts (the extreme right). This is illustrated in Figure 17 through the use of arrows which are directed from the bottom left corner to the top-most right corner, and vice versa.

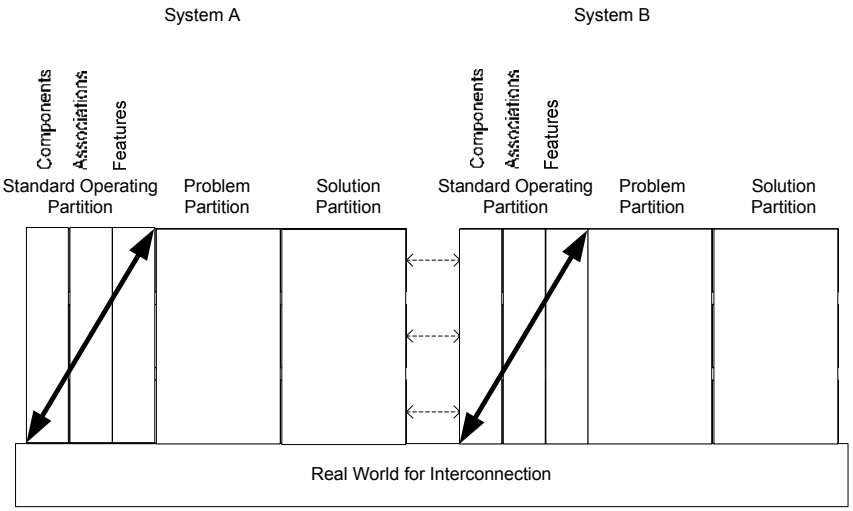

Figure 17 - The inclusion of minor-partitions follows an abstraction paradigm
Abstraction of details across the minor partitions should be from most specific to least specific in order to facilitate abstraction across system layers.

The reason for this depiction is now explained. If one was to examine a major partition which had been decomposed using minor partitions, with the inclusion of layers, it would be apparent that in examination from the real world layer's bottom left-most corner, to the strategic layer's top-most right corner, a diagonal direction would be followed. This convention is intended to convey the degree of abstraction of RFID systems. That is, some systems abstract the environment more in terms of the number of layers between the real world layer and strategic layer – the Electronic Product Code (EPC) system which uses the Object Name Services (ONS) is one such system (Ranasinghe et al. 2008) – the ONS is a data layer which fits over the RFID. Thus, this model has a strong representational basis for actual RFID systems – a feature which was not facilitated by previous work examined in Chapter 3.

Consequently, this could mean that consideration can be given to whether a threat is dealt with close to the source or destination. For example, systems which exhibit

many layers and partitions between the source of attack and the attack destination, could offer more points for security solutions, in addition to more points of attack. This approach to abstraction is a feature unique to this model when compared to previous work.

To briefly summarise, the use of a vertical *partition* property strongly distinguishes this reference model from previous work. Partitions enable the separation of independent but related security concepts – standard operations, threats, and solutions. The advantage imparted is that security is analysed across three major concepts in a system, making analysis more structured than if just threats or solutions were to be considered without relation to one another.

5.2.3 INTEGRATING LAYERS AND PARTITIONS

The *integration* of layer and partition properties enables a closer comparison between the RFID system and RFID security. This was a major shortcoming which was apparent in previous work which was reviewed in Chapter 3. This section explains how integration addresses this shortcoming, and consequently, makes possible a 'whole of system' approach to the analysis of security in RFID systems.

The integrated layered and partitioned reference model is reintroduced in Figure 18, in its complete form. In this diagram it can be seen that the layers have been divided using the partitions. This is where the integration of these properties is depicted. Although there is a small separation between the layers, the partitions span all the layers, thereby enabling the derivation of security concepts under a single partition, across all three major layers.

To this end, the *standard operating partition* should be thought of as spanning the *strategic layer*, *RFID layer*, and *the real world layer*. The minor partitions have been included in the diagram; however, the use of these minor partitions is only illustrative of this integration concept – minor layers are in effect optional. Similarly, the use of the OSI layers at the RFID layer has not been depicted; however these could be added when the model is applied to real analysis problems. The rest of this section expounds the reasons for this integration.

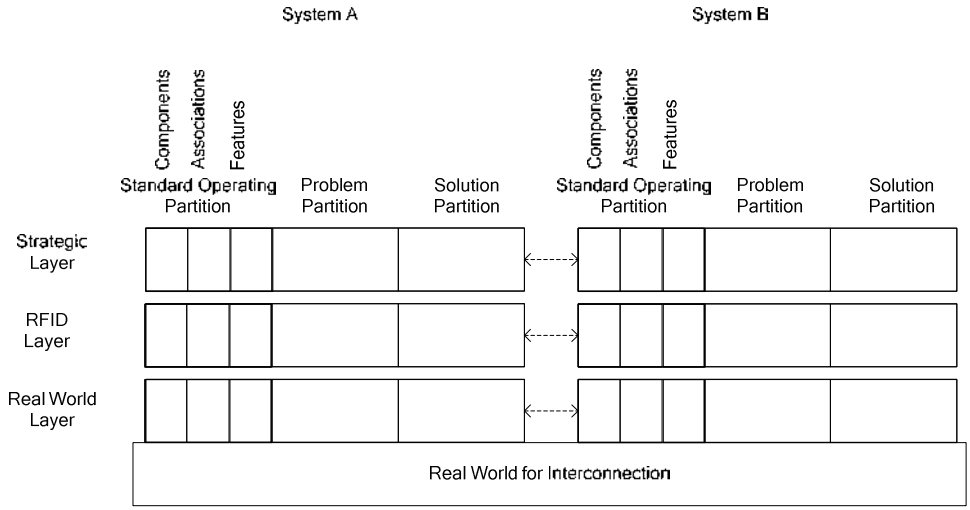

Figure 18 - The integrated layered and partitioned reference model
The model is reintroduced in this section having explained the properties of layers and partition separately.

Integration makes it possible to analyse partition concepts across layers of an RFID system. For example, the standard operating partition can model: the physical entities and physical constraints at the real world layer; at the RFID layer (the RFID components such as tags and readers, anti-collision and other protocols); and at the strategic layer (the information goals of the company whether these are monitoring or authorisation). Integration enables 'whole of system' analysis in each partition across system layers.

Through the integration of both properties, it is possible to consider RFID and security concepts collectively. For example, the standard operating partition can be used to consider how the strategic layer information goals of monitoring or authorisation are being facilitated by the RFID components, and whether these components are integrated within the real world layer adequately to achieve these information goals. For the other partitions, the same approach can be used to take a systematic view of the concepts they represent.

An approach which considers interrelationships is enabled as the major partitions are aligned to each other. Comparisons can be made between partitions but at different layers. For example, the solution partition could model solutions at the strategic layer, and these could be compared to the problems which occur at the same layer of the problem partition. Whether modelled solutions and problems are in fact feasible or relevant to a system could be ascertained by modelling the standard operating

partition at the strategic layer as well. Thus, integration of layers and partitions facilitates a 'whole of system' approach to the analysis of security in RFID systems.

It is therefore possible to compare partitions to each other across all of the system layers. Integrated security analysis is therefore achievable. Questions can be proposed and answered along the lines of:

- How can solutions at the strategic layer address threats which occur in the RFID layer in an actual system?
- How will the choice of anti-collision protocols influence which threats may be possible in a system?
- How will a system's real world layer influence which attacks are feasible in the RFID layer?

To this end, the integration of layers and partitions enables effective security requirements analysis to be undertaken. It would be possible to consider security questions which are created across all facets of the system rather than on individual components.

5.3 SUMMARY

This chapter has introduced an alternative model for the analysis of security using a 'whole of system' approach. It was constructed using the reference model paradigm reviewed in Chapter 4. It is distinguished from previous work (see Chapter 3) on the basis of integrated layer and partition properties, and has therefore been entitled, *An Integrated Layered and Partitioned Reference Model.* The integration of these properties is expected to be more conducive to a 'whole of system' approach to analysis when compared to the examples of previous work.

Recall that there were some apparent limitations on previous work which the design and intended use of the proposed model addresses. The model proposed by Rotter (2008) does not appear to be suitable for analysis beyond several system properties. Conversely, the model proposed by Mitrokotsa et al. (2010) allows for the use of security principles and attributes only within individual layers, thereby reducing its generality.

As it seems that previous work has been localised to specific system properties, which has the drawback of missing the interrelationships which are relevant throughout the 'whole system,' the concept of integrating layers and partition properties was introduced in the proposed model.

The horizontal system *layer* property captures system components at a relatively high level of abstraction. One advantage is the ability to encapsulate previous work within representations of system layers. The organisation of these layers, with the RFID layer in the centre, suggests that RFID is a central concept, but should be viewed in conjunction with the real world layer and the strategic layer. Conversely, the *partition* property facilitates a means of analysing security in the RFID domain. Partitions demarcate the domain such that solutions can be evaluated against particular threats, in particular system contexts. The organisation of layers, from the least abstract to the most abstract imparts the principle that analysis is based on what a system functionally supports. Attacks and solutions are then related to the systems functions.

The integration of layer and partition properties provides an option to use the model for complete 'whole of system' analysis of security. Whether analysis has proceeded per layer or partition, the outcomes of analysis through the model should mean that an effective view of security can been taken. The discussion in Chapter 2 suggested that could be a desirable approach to take, as cloning and constraints have system wide influences. Integration allows all of these interrelationships to be taken into consideration during analysis.

The following chapters will illustrate that when this model is used, structured security analysis can be achieved:

- Chapter 6 illustrates how analysis using the standard operating partition facilitates the enumeration of system elements into a domain model.
- Chapter 7 illustrates how analysis using the problem partition enables systematisation of attacks.

- Chapter 8 introduces a simulator model and demonstrates through its use the benefits of taking a systems approach to solution analysis in the solution partition.

- Chapter 9 demonstrates, through experimentation, how the model facilitates systems analysis prior to solution deployment.

- Chapter 10 validates the 'whole of system' approach in the context of the specific example of a pharmaceutical supply chain.

The work presented in these chapters will show that the reference model improves on previous work by providing more structured security analysis.

Chapter 6

The Standard Operating Partition and a Domain Model

This chapter is partially based on a publication presented at the 5th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne, Australia, 2009 (Mirowski et al. 2009c)

This chapter is also partially based on a publication presented at the 10th International Symposium on Pervasive Systems, Algorithms, and Networks, Kaohsiung, Taiwan, 2009 (Mirowski et al. 2009a)

6.1 INTRODUCTION

In order to facilitate analysis of security 'whole of system', the broad constituent elements of the system should first be established. In this chapter, this is done within the *standard operating partition* of the reference model. This partition is for modelling the standard operations of RFID systems: How this can be achieved is explored when generic elements of the system's internal boundaries are enumerated as a domain model. As a 'whole of system' approach is applied to this task, it is necessary to make use of methods which are capable of enumerating system elements in a structured manner across system layers. Pressman (2000) argues that if you want to use a systems approach, you should start with a model which is a representation of the processes, behaviours, inputs and linkages of the system in question – this is the area to be explored in this chapter.

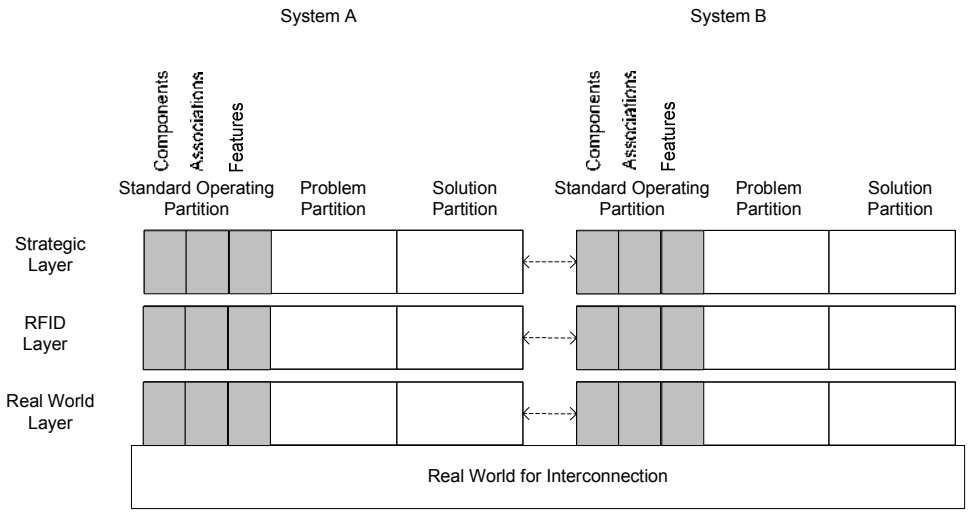

Figure 19 – Analysis of the standard operating partition
This chapter illustrates how a domain model can be produced when systematic methods are applied across the layers within this partition.

In Chapter 3, in addition to the architectures proposed by previous work, some work had identified components at various layers. Work by Hassan and Chatterjee (2006) illustrated a taxonomy of system components and sub-components, whereas work by Glover and Bhatt (2006) represented some RFID and other components. When considering these examples it seems that they have identified various objects and where they are located in system, and their relation to other components; however, they have not appeared to identify the standard operations of objects. These

operations are the functions each object initiates to perform its role in the system. One must understand how interrelationships emerge between components, through component operations and interactions, in order to understand security 'whole of system.' Conversely, work on security models has discussed general system components but not applied this analysis in a domain model. Therefore, integrating a model of the domain within the reference model would provide a basis for analysing security in a systems context.

To this end, in this chapter the standard operating partition and its minor partitions are analysed and this is illustrated in Figure 19. It defines a logical view of components by focussing on the major system components, their attributes, and operations. As RFID is a system which produces data, when components have interacted, it will also define a model of associations and features, thereby providing a data view. This work will introduce the concept of an RFID domain model, and in the later chapters, will use this when analysing the other partitions of the reference model.

6.2 COMPONENTS PARTITION

In this section, the standard components which are common to RFID systems are enumerated using Object Oriented Analysis (OOA), and these are modelled using the Unified Modelling Language (UML) in a class diagram (Bruegge and Dutoit 2004; Maciaszek and Liong 2005). As described in Chapter 5, the *component partition* is the demarcation within the reference model's standard operating partition which contains the logical components of the RFID domain. The major RFID components considered here are: *tags*, *readers*, and *database* (Glover and Bhatt 2006). Several additional components are included: *zone* and *physical entity*. These have emerged from previous work which has considered a broad view of systems (Mitrokotsa et al. 2008, 2009). They are included to achieve a broader view of the system at the real world layer and strategic layer.

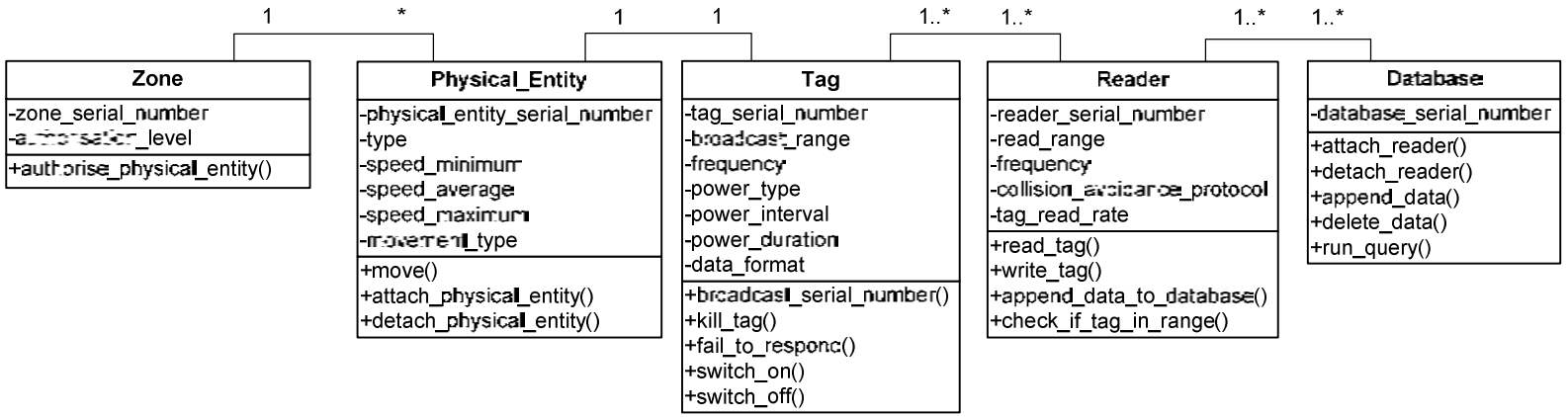

Figure 20 – UML class diagram of the major components

Each component of the system is represented as a discrete object, having its own set of attributes and operations. The model is extensible so future work can build upon the classes it encapsulates

Using the UML notation for *class diagrams*, each type of RFID component is represented as an *object class*. The properties of a component have been generalised; these describe properties found as common to a number of actual systems. The focus is on the physical representation at the layers in which these exist. The process in deriving the generalisations was driven by the question; what constitutes the object? The constituent parts of an object are represented as: an *attribute* which contains state values of a component; and *operations* which encapsulate functions that a component can perform to derive a new attribute value. For example, every component in a system should have a *name* attribute which is unique to the RFID system – a tag serial number is one such name value common to tag components. An example of a valid *operation* for a reader is the *read* operation – when the reader queries the environment for tags. In the case of tag cloning, a tag could derive a new name when an attacker reprograms it with a different serial number – this would be achieved using a *clone* operation. The rest of this section expands on the class diagram.

6.2.1 COMPONENTS

This section describes the major components in their standard operational form. Figure 20 shows the class diagram for the five components of the component partition. These components are: *zone*, *physical entity*, *tag*, *reader*, and *database*. They were selected across the three major layers of the reference model to show the hierarchical operations that lead to the production of output data. Other components could have been considered, but were not, such as antennas, anti-collision protocols, and other middleware. These are depicted in representations, such as those by Hassan and Chatterjee (2006), however, the major components here are chosen to focus on the major characteristics.

6.2.1.1 ZONE

A *zone* (Figure 21) represents the physical space within which an RFID application is situated. Often the concept of a zone is referred to as the entire application environment; however, in this section a zone is introduced as a single location which is monitored by RFID components. This distinguishes it from the *application environment* which contains a number of zones, and also without RFID.

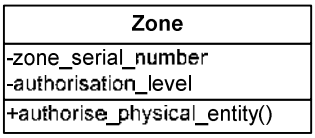

Figure 21 - Zone component

Numerous examples of zones exist in the literature: a dock door that is inside a warehouse; a toll gate along a toll road; a doorway inside a building; or abstract space around components situated on a forklift in a warehouse (Glover and Bhatt 2006). In addition to containing RFID components, usually a zone will constrain the movement of components in some way. A doorway, for example, could only open in one direction. Additional properties could be derived by a zone, from other zones, which introduce further constraints. A doorway, for example, could be only accessible via a series of other doors, or the RFID system could impose restrictions on individual tags based on the serial number contained on the tag and permissions of a policy database.

To this end, a zone is constituted by a number of attributes and operations which reflect its role for spatial demarcation. A zone is typically distinguishable from other zones within an overall zone through a *zone_serial_number* attribute which is its unique name. For example, different warehouses in a supply chain could have their own names, or different sectors in a building can be named differently. If the zone specifies a permission policy then an *authorisation_level* attribute can be used to dictate the level of authorisation physical entities, and hence their tags have to attain in order to gain access to it. In this section, a zone is assumed to perform a single major role, in that it enforces policy for the strategic goals of the system.

The concept of authorisation is modelled as the *authorise_physical_entity* operation. This represents the checking of entity credentials, perhaps through RFID tags which identify the physical entity to the system, and also the actuation of mechanisms which enforce authorisation such as locks on doorways. In this case, a zone works in conjunction with an RFID system to establish the presence of a physical entity through tags, establishing the permission of that tag through the reader, and associated permissions stated by the company, prior to it enforcing such access controls.

6.2.1.2 PHYSICAL ENTITY

A *physical entity* (Figure 22) represents a *physical object* in a zone. A physical entity could be an object which is of interest to the system, for example, a product or human. This concept comes from various examples which have considered the system to be more than just RFID technology. It could be a component which is used to constitute a larger component. The ingredients in pharmaceuticals would be an example of this object relationship. Conversely, it could be some object in the application environment which is of no direct interest to the system, however which in some way influences how other objects work. The primary interest for RFID security should be limited to entities directly influential in the RFID system.

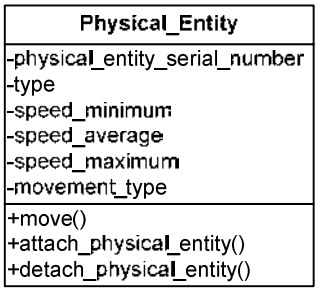

Figure 22 - Physical entity component

As the physical entity represents a tangible object in the application environment, the attribute and operations reflect its physical characteristics. A physical entity's attributes are used to establish an entity's profile in a system. As a system could contain many physical entities, the concept of an identifier, the *physical_entity_serial_number* attribute, is used to distinguish physical entities. It could be possible to distinguish physical entities based on existing names. For example, people in a system could have names which identify them, and similarly pharmaceutical products contain a product name as well as a manufacturer name such as the Trizivir drug manufactured by GlaxoSmithKline (O'Connor 2006). However, the desire to use RFID in systems indicates that some systems otherwise lack the ability to assign such identifiers, or in the least have the system establish them, as the entity is not capable of communicating the name to the system.

The *type* attribute is used to specify the entity class from which the entity is instantiated. Some systems could instantiate entities from a common template – pharmaceutical drugs are one such system, as many drugs are manufactured to a

single specification, a batch, or dosage strength. The concept of a class defines a range of specificity for instantiated entities. As the physical entity is assumed to be capable of acting independently of other objects, other attributes model the ability to move in the system: *speed_minimum*, *speed_average*, *speed_maximum*. Also, a *move_type* attribute models whether a physical entity could move in a deterministic or stochastic manner.

The operations of a physical entity model the valid movements within zones. The *move* operation models how the physical entity can move from one location towards another location using the values instantiated in its speed attribute. The exact underlying steps involved in moving are hidden away in this operation; it is sufficient to model that there is a concept of movement. A physical entity could also be associated or disassociated with other physical entities, and hence move in conjunction with them, which is reflected in the *attach_physical_entity* and *detach_physical_entity* operations. These operations would therefore signify the composition or decomposition of several entities, and thus, whether entities were affected by each other's behaviour. A group of products on a pallet is an example of several disparate entities being combined to form a single entity, in which case, the products would inherit the movement behaviours of the pallet.

Thus, individual physical entities form the conceptual building blocks of all objects within an RFID system, while the concept of zones represents the environment in which these physical entities exist. This establishes that the RFID system is not some ethereal system – rather it is grounded in a physical environment, and hence, subject to the rules governing the environment. In addition, this is why the reference model includes a separate *real world of interconnection* layer and *real world* layer, and hence, this component identifies objects at these layers.

6.2.1.3 RFID TAG

A *tag* (Figure 23) represents an electronic device that acts as a surrogate for a physical entity by broadcasting a unique serial number using radio signals (Glover and Bhatt 2006). It is distinguished from other automatic identification technologies, like barcodes, on the basis that it communicates a unique serial number or identifier and does this using radio signals and without requiring line of sight.

Although some RFID readers can perform tag operations, this model distinguishes a tag from a reader as a device which simply broadcasts data into the air. The focus is on passive tags which are powered using passive methods. Tags of this type include the Electronic Product Code (EPC) Class-One Generation-Two type (EPCglobal 2005).

Other types of RFID tag exist – active tags are one such tag type – however, these are not as easy to generalise, as what constitutes an active tag can be difficult to define. Mobile phones, for example, could be a type of active tag, as they use an onboard power source and they use radio signals to broadcast an Extended Service Set Identifier (ESSID), which is a type of serial number. Mobile phones these days also come with Near-Field-Communication (NFC) technologies which enable them to act as short range RFID tags and readers simultaneously. To maintain simplicity, the model is limited to passive tags; however, as the model is extensible, active tags could be modelled with more investigation.

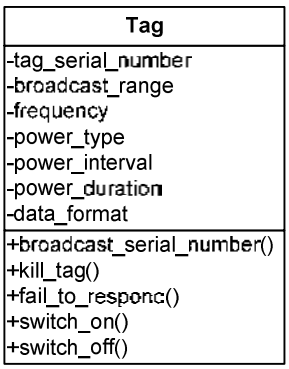

Figure 23 - Tag component

The attributes of a tag reflect its surrogacy role in RFID systems. For most RFID systems, it is important that every tag is unique, and therefore, in the model a tag has an attribute which stores a unique serial number, the *tag_serial_number* attribute. Another name for the serial number is the Electronic Product Code (EPC) (Finkenzeller 2004), however, this is not used here as it is synonymous with the EPC standard. A tag operates at a particular radio *frequency* to obtain power from a reader. Whether it is powered by a reader or by an onboard power source, this is specified in the *power_type* attribute. As this model is not capturing active tags, this could be used to signify that a tag is of interest as it is passive not active. For passive

tags, those which obtain their power from a reader could be subject to a charge and discharge cycle which controls how quickly they dissipate power received from the reader. To model this, a *power_interval* attribute is modelled. To signify the maximum distance a tag can send back its tag serial number, a *broadcast_range* attribute has been modelled. Clearly this simplifies the complex nature of radio frequency and the effect of environmental factors; however, including this attribute simplifies how a user conceptualises RFID under ideal conditions. The last attribute, *data_format* represents the method by which data has been encoded in tag memory.

Usually all tags are concerned with identifying themselves to the system using radio signals and unique serial numbers. The variance in tag types comes from the ways in which these concepts are established in the system. Passive tags achieve these by obtaining power from a reader, however consequently, are shorter range than active tags (Glover and Bhatt 2006). Conversely, active tags have a greater read distance than passive tags, and can perform more complex operations. To sidestep these complications, the model focuses on the principles which tags encapsulate. A *broadcast_serial_number* operation models that a tag listens for requests and responds via its radio signals and a modulation scheme to a reader with its data. However, to undertake this operation, a tag has to be powered, in which case it will *switch_on* when it has sufficient power from a source. Conversely a tag could *switch_off* once power has been dissipated. These operations indicate the time period during which data can be obtained from a tag.

Some operations are unique to some tag types and are briefly mentioned here. As passive RFID is an imperfect technology, transmission errors or environmental interferences can disrupt transmission, a tag could fail, in which case the concept of a *fail_to_respond* operation models how errors are handled. Sometimes tags can fail on purpose. Some tag types such as the Electronic Product Code (EPC) come with an inbuilt self destruct command (known "Kill command" in EPC systems) which can permanently deactivate a tag when it is no longer needed for use, or for security reasons (Glover and Bhatt 2006) – this has been represented as the *kill_tag* operation.

6.2.1.4 RFID READER

A *reader* (Figure 24) represents an electronic device in an RFID system that powers tags via radio signals – if it is a reader for passive tags - and also captures tag

responses over the air (Glover and Bhatt 2006). A reader can also transfer received data, in the format of *data records*, to a database, usually located in the middleware. The concept of a reader is essentially a device which bridges the real world layer to the RFID layer as it performs these duel roles of reading tag data and transferring data to the middleware. Sometimes readers can perform additional tasks such as *filtering* or *aggregation* of RFID data – a role which sees them remove duplicate records obtained in time windows to avoid overloading databases with data - however; these additional operations are not included here to ensure the model remains simple. Sometimes these operations occur in other parts of the system – for example, Glover and Bhatt (2006) illustrated that filtering could occur in the middleware. In general, a reader's main operations are modelled, and other operations are left to future work.

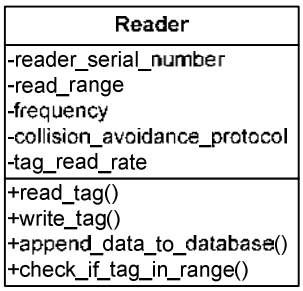

Figure 24 - Reader component

To model the reader's apparent duel role as the arbitrator of tag responses and a source of information for databases located in middleware, the reader has attributes which distinguish between these roles. The *reader_serial_number* attribute identifies the reader from the other readers to which tags could interface when in range. It could be that this identifier is also used to identify the reader on the network to databases. The reader broadcasts signals at a particular radio *frequency* to identify tags of the same frequency. News sources report that some modern readers, such as the *ThingMagic Mercury,* employ software defined radio capabilities and this means that some readers are more flexible in terms of the frequencies they can read (Collins 2004). In such a case, this model would require additional fields, or a single field which signified the various frequencies across which these can operate.

If the reader has the capability to distinguish several tags which are in its range simultaneously, then the reader will be using an anti-collision protocol (Glover and

Bhatt 2006). The *collision_avoidance_protocol* attribute can be used to specify which algorithm is employed. As a derivative of a reader's anti-collision protocol and frequency, amongst other properties, the *tag_read_rate* attribute is the way of modelling the maximum number of tags that could potentially be identified within a time period by the reader. It simplifies a series of complex operations, but would be a useful means of indicating a reader's performance.

The operations performed by a reader are contained in several relatively high level operations. The *check_if_tag_in_range* operation represents the propagation of a carrier signal and capturing of tag responses. Acquiring data from a tag is represented in the *read_tag* operation, as a tag could be powered, however, ultimately its appearance in a data log is at the discretion of the reader. By no means does this capture the intricate set of steps which are performed by modern protocols such as the Class-One Generation-Two protocol (EPCglobal 2005); however, it does capture the final effect of a successful tag read, i.e. data is produced.

The *write_tag* operation represents the act of transferring data to a tag. To signify a reader's ability to send any captured data from tags to the middleware, usually to a database, the *append_data_to_database* operation represents the act of transferring that data to a database. Again, this could be a complicated process as databases could be located over vast networks like those which have been proposed in the Electronic Product Code (EPC) specification. In this case, additional middleware services are involved, like the Object Name Service (ONS), in order for remote databases to be located, and EPC data records to be retrieved and updated. The exact implementation of such operations is encodable in the operation, however, it remains hidden at this high level of abstraction.

6.2.1.5 DATABASE

The concept of a *database* (Figure 25) is the last component which has been modelled in the component partition. It is perhaps the most abstract component, as it represents the repository of information which is produced from the RFID system. Databases can be complex components, however, in this model the role of the database is relatively simple – data flows into the database from the reader, and other components, perhaps those which exist in the middleware, retrieve data from the database. This representation is taken from work by Glover and Bhatt (2006), which

modelled it as the destination of information flowing from tags and readers. The difference is that this model does not discriminate where in the layers it may be located, as it could be at the RFID layer or the strategic layer depending on the level of information sharing in the system.

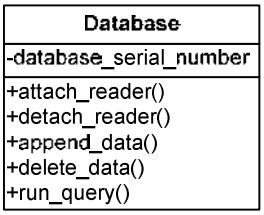

Figure 25 - Database component

As the model has assumed that a database is concerned with the storage and retrieval of data from RFID readers, readers must be able to be associated with a database. To this end, the *attach_reader* operation is a way of connecting these components, while a *detach_reader* operation represents disconnection between these components. Sometimes these operations could be used by multiple readers as the database could be connected to multiple readers in a networked environment, for example. To represent processes which could work on the data which is stored in a database, a *delete_data* operation and a *run_query* operation are modelled. The exact nature of these operations is not specified as there could be any number of queries which could be performed. Essentially these all represent ways in which information about the tagged entities in the real world by components above the database can be sourced.

To summarise, this section has introduced a logical view of the major components which broadly constitute the *component* partition. Using these generic object classes, depicted in UML, an RFID system could be modelled at a relatively high level. The model focuses on standard operations as a way of describing what component functions are in a system. This enables understanding about how interrelationships emerge when components initiate operations or interact with each other. The next section considers the result of interactions in the logical view, which consequently, lead to RFID data being produced.

6.3 ASSOCIATIONS PARTITION

The *component* partition contains the object classes from which components are instantiated. The enumeration put forward in this chapter represents the objects as static constructs and consequently does not give any indication when these objects interact. For RFID systems the concept of interaction between components is an integral process - interaction of tags and readers produces RFID data which is the basis for strategic level decision making. It is the RFID data which is used by the system owner, e.g. a company, to determine if certain events have taken place in the real world. However, before it is possible to think about what can be inferred from RFID data which has been produced when tags and readers have interacted, the next step is to consider the ways in which tag and reader components can interact.

This section models the *associations* between tags and readers in the *association* partition. It models potential ways in which associations may emerge at the RFID data layer, between RFID components. As the data is generally assumed to be produced when a tag is at a reader, the associations which are modelled are representative of associations which have formed between a tag and reader, and at the layer below this, the real world layer, between entities which have tags and readers attached to them. As the data is being modelled in an RFID database, this section applies the principles of *entity-relationship modelling* first expounded by Chen (1976) to model these tag/reader associations to define a *data view*.

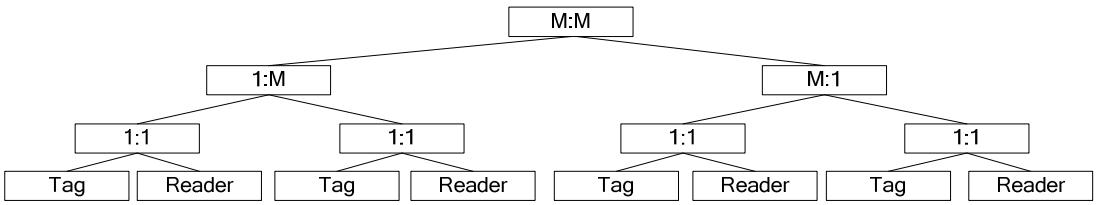

Figure 26 - RFID tag and reader associations
These have been modelled using ERD multiplicity concepts. A one-to-one (1:1) association is when one tag interacts with a single reader; other associations are built up from a 1:1 association when additional tags or readers are introduced or when anti-collision is used at a reader. Thus, a hierarchical model is introduced to represent the ways that associations can form.

Figure 26 introduces the concept of a taxonomy of tag and reader associations which can be found in RFID data which characterise underlying tag and reader relationships. The rest of this section adapts the concept of *multiplicity*, in particular

the four standard multiplicity relationships (1:1, 1:M, M:1, and M:M), which exist in entity-relationship modelling, and shows that it is possible to model associations between tags and readers using the same multiplicity relationships. The knowledge imparted through the use of these terms is a controlled vocabulary to describe RFID structures – a key concept useful in allowing end users to identify domain abstractions (Arango 1994).

6.3.1 ONE-TO-ONE (1:1) ASSOCIATION

A *one-to-one* (1:1) association, illustrated in Figure 27, is instantiated in RFID data when contact is made between one tag and one reader. For example, a tag has responded to a reader's *read* command, in which case, data has been obtained from the tag. For the Mobilkom NFC payment system, which allows a mobile telephone to communicate with a single NFC-enabled cashier terminal to pay for goods (O'Connor 2005a), as the NFC-enabled telephone establishes a connection to only one cashier, and the cashier is capable of only one NFC telephone connection at a time, this is an example of a 1:1 association. Each time an association is formed, in this case, an RFID data record containing a timestamp and the component serial numbers is instantiated at the reader.

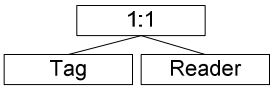

Figure 27 - One-to-one (1:1) association

The rest of the associations build on the concept of a 1:1 association between a tag and reader, but introduce more facets of the system which are available to the database if it was to look across multiple data records and data sources.

6.3.2 ONE-TO-MANY (1:M) ASSOCIATION

A *one-to-many* association (1:M), illustrated in Figure 28, is instantiated in RFID data when a tag has made contact with several readers in sequence in the system. The tag has been *read*, for example, at a reader and has then moved to another reader which in turn has also read the same tag. As these readers share a common database where they store their data, the database would be able to infer from the supplied data records, that the tag has been engaged with several readers over time.

An example of a 1:M association having been formed can be found in the Orlando/Orange County express way toll system which monitors vehicles via an RFID tag as vehicles travel along roadways which have readers mounted at certain locations (Swedberg 2004). As a vehicle uses a tag, which contains a unique serial number, over time this tag would have appeared at several readers in sequence, as the vehicle travels along the roadway.

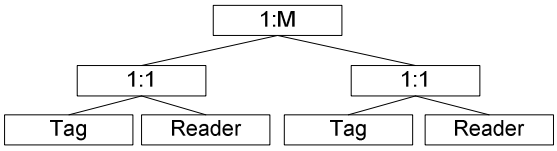

Figure 28 - One-to-many (1:M) association

In this association it is important to recognise that at each reader a 1:1 association is formed, however, from the perspective of the database which has knowledge of all the readers, a 1:M association can be inferred. This is why the model shows associations in a hierarchical representation, as this representation depicts the concept that more complex associations are based on the basic 1:1 association.

6.3.3 MANY-TO-ONE (M:1) ASSOCIATION

Conversely, if a reader had several tags in front of it, and the reader is using an anti-collision protocol – these enable multiple tags to be read simultaneously (Finkenzeller 2004) - then it is possible that a many-to-one (M:1) association is be formed. This is illustrated in Figure 29. The anti-collision protocol schedules each tag in the reader's vicinity to respond according to a scheduled time (*Aloha*-based protocols) or when addressed individually through their tag serial numbers (*Tree Walking* based protocols). Instead of every tag responding at the same time period, tags will respond according to how they have been scheduled, thereby avoiding collisions. But the reader could examine all records within a time window at a reader to determine which tags were active, and hence, constitute a M:1 association.

An example of this is the Dalsey Hillblom Lynn (DHL) Smart Box, which allows a physical entity to be identified via a tag when tagged entities are placed inside a container which contains a reader (Wessel 2007). There are many tagged entities and only one reader in this example and is thus a M:1 association.

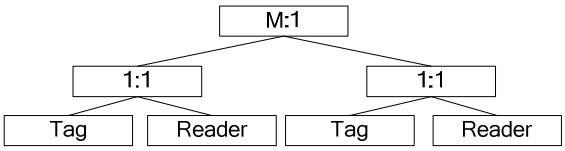

Figure 29 - Many-to-one (M:1) association

As an anti-collision protocol allows many tags to communicate with a single reader at a time, conceptually, it is possible for a database to treat all tag data records as about a collection of tags, which in this section is termed a *tag group*. The formation of a tag group could be considered to be random or non-random. A random tag group is when an unrelated set of tagged entities arrives at a reader, such as when a group of vehicles congregate at a tollgate. Conversely, a non-random tag group is formed when products are arranged purposefully, such as when products are arranged on a pallet or in packaging. Recognising that tag groups are phenomenon reflected at the RFID layer through tag signals and also in RFID data, may allow for more effective recognition of groups in the real world.

6.3.4 MANY-TO-MANY (M:M) ASSOCIATION

A *many-to-many* association (M:M), illustrated in the full model in Figure 30, is instantiated in RFID data when a tag is read across several readers which are using anti-collision protocols, and thus, could have read other tags at the same time as this tag. The M:M association subsumes all previous associations, and is thus, represented as the top-most association in the model.

An example of where a M:M association can be found is in the International Paper RFID system which monitors physical entities when they are placed on a forklift (O'Connor 2005b); as there could be many tagged entities on a forklift and many forklifts with readers in warehouses, this is an example of a M:M association.

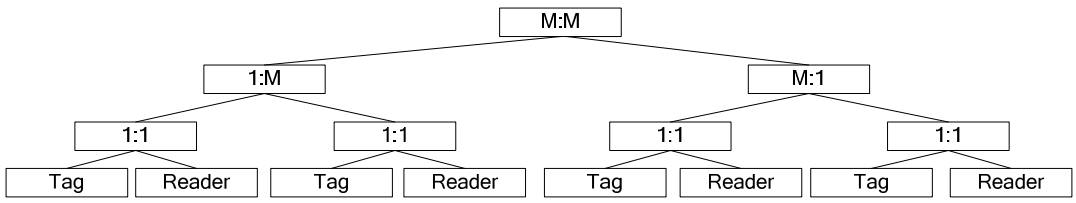

Figure 30 - Many-to-many (M:M) association

A M:M association subsumes previous associations, while at the same time the structural properties of the system which have been built up can be decomposed to the most elementary 1:1 association. At the elementary level each tag is communicating with a reader through a 1:1 association. At the level of the M:1 association, anti-collision places each tag into a collection of tags read simultaneously. As a tag moves through a system, it could become a temporary member of tag groups at readers, or could already be in a fixed collection of tags. This movement through the system constitutes a 1:M association, while conformance to a collection of tags is a M:1 association. At various times in the system, different associations are formed, and different information can be gleaned from tags. The ability to glean such information is related to the system facilitating the establishment of these structures.

Finally, within the taxonomy a controlled vocabulary has also emerged as a way of describing the clusters of object classes that can be identified. Recall that Arango (1994) suggests that the derivation of a controlled vocabulary is beneficial to a domain model. It assists one to understand the domain at an object and data level. The concept of having 1:1, 1:M, M:1, or M:M groups of tags and readers is supported by a description of how they are formed through spatial and temporal enablers.

To summarise this section, what has essentially been described is a taxonomy of associations between tag and reader entities, enabled through the real world layer, but visible at the RFID layer in RFID data. The taxonomy abstracts classes of entities, like that in an OOA diagram, as at one end very specific object classes can be identified, to the other end where very general object classes can be identified. These are visible at the data level due to the hierarchy and information flow in RFID systems. These structures identify existing formations of tags and readers in systems - the advantage imparted here is the formalisation of these in a model. Knowing these structures exist, and what they contribute, assists users in realising their effects in systems.

6.4 FEATURE PARTITION

Having described the major components in the system, and the associations which can be formed between tags and readers from a database's perspective, this section examines what information may be constructed from these associations.

In this section features are derived for the above four tag/reader associations using a feature construction approach. The approach taken can be likened to a knowledge driven approach (Wnek and Michalski 1994; Alfred 2008) where typically, features are formed on the basis of knowledge of associations between existing features. As these approaches emerge from the machine learning domain, the term feature construction is maintained, although a systems analysis approach is taken here and the term is not necessarily common to this domain.

In this section, the feature construction process begins when the simple RFID data features formed by a 1:1 association, are transformed into more complex features on the basis of associations built up from the 1:1 association. For example, in a system which contains a 1:M association, whereby an instantiated tag is read across multiple readers, a feature which could be constructed is the *speed* at which the tag has travelled between these readers. New features are constructed, and then they are inserted into the feature set for each association and used to construct further features. The purpose of this is to expand the model of the domain in the standard operating partition's data view.

6.4.1 ONE-TO-ONE FEATURES

Table 2 shows the elementary RFID data features common to tag and reader interactions (Finkenzeller 2004). This thesis links these elementary features to the 1:1 association as this association represents a single point of contact between a single tag and reader. When a tag and reader have established a 1:1 association, these features would be instantiated with values for the tag and reader involved in the association. Every time a tag is read, for example, a new tuple containing values for these features is instantiated inside a database for an RFID system.

Table 2 – Elementary features of a 1:1 association

Feature	Meaning
tag_serial_number	Serial number or identifier of tag such as its Electronic Product Code (EPC).
reader_serial_number	Serial number or identifier of reader
session_timestamp	Timestamp of when tag and reader interacted, such as through the *read* command.
reader_operation	The particular RFID reader operation performed on the tag such as a *read* or *write*.

Table 3 shows the features which have been constructed when the elementary features of a 1:1 association are viewed from the perspective of a database. As the database has a wider system view, it may aggregate individual data records to form new features. For example, the *total_number_of_tag_reads_at_reader* feature could be constructed simply by counting the number of data records produced by a tag at a reader, although this would involve the database trawling its history of data records to achieve this goal. Such a feature could be used to indicate the frequency of use of a particular tag at a reader. These features pertain to an individual tag and reader instantiation and do not require any other component instances; therefore, they are still 1:1 features. [2]

Table 3 – Features of a 1:1 association

Feature	Meaning
first_time_tag_seen	First time the tag was read at any reader in the RFID data.
total_number_of_tag_reads_at_reader	Number of *read* operations, and hence data records produced, when tag and reader have interacted.
session_duration	A *session* could be a clear time frame when a tag was active at a reader. For example, all data records produced within a 10 second time window could constitute a session.
time_since_last_seen_at_this_reader	Time between successive data records being produced at same reader.

As indicated in the association partition of the previous section, associations are hierarchical in nature which is why features which have been derived for subsequent associations from 1:1 associations. These are discussed in the next sections, and are essentially compound features which arise when features are combined with spatial

[2] Some of the feature tables here have been modified since their original publication in Mirowski, L., J. Hartnett and R. Williams (2009a). How RFID Attacks are Expressed in Output Data. Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN 2009), pp. 794-799.

or temporal characteristics made available by additions in the system such as the addition of different combinations of tags or readers, or anti-collision protocols – all of which are characteristics of the various associations.

6.4.2 ONE-TO-MANY FEATURES

Table 4 shows the features which have been constructed for a 1:M association. From a database's perspective, as it may have knowledge of a several readers, these features utilise its knowledge of the spatial distribution of readers in the system. Provided some data records have already been produced, the database could query the historical records to determine a tag's *speed*. To this end, several features which measure speed have been constructed. The *speed_of_tag* feature for a tag would be calculated by calculating the time between two data records produced at different readers and dividing this value by the total distance travelled (provided this distance was known). Another feature is the *read_direction_of_tag* feature. If the sequence of readers is known in the system or the direction a tag can follow is directed, then it is possible to determine the direction value as *forward* or *backward*, for example. In a supply chain, such a feature could be useful to indicate whether a tag, and hence the attached product, was moving in the correct direction.

Table 4 - Features of a 1:M association

Feature	Meaning
first_reader_tag_read_at	Reader a tag first read at in the sequence of readers.
current_reader_tag_at	Reader a tag currently being read at
time_between_all_readers_for_tag	Time between tag *read* operations at first reader in the sequence of readers to the current reader.
time_between_same_tag_at_different_readers	The time between successive tag reads at current reader and a previous reader. E.g. subset of readers in the sequence.
average_time_ between_readers_for_tag	Average time between successive tag *read* operations at different readers.
reader_order_for_tag	Sequence of readers a tag was read at
read_direction_of_tag	Direction through a sequence of readers a tag travelled. Either forward or backwards, provided the underlying application environment is directed.
speed_of_tag	Speed a tag travelled at, between current reader and a previous reader (assuming the distance travelled is known by the system).
average_speed_of_tag	Average speed tag travelled between current reader and a previous reader.
total_distance_travelled_by_tag	Total distance tag has travelled between a sequence of readers
average_session_duration_ for_tag_at_all_readers	Average duration of a read session across several readers
previous_reader_tag_read_at	The reader a tag was previously read at prior to this observation.

The above features introduce the concept that knowledge of the locations of readers by the database allows it to construct features about an entity's movement in the system.

6.4.3 MANY-TO-ONE FEATURES

Table 5 shows the features which have been constructed for a M:1 association which is when many tags have been identified at a single reader at (effectively) the same time. The *group_read* feature is the collection of *tag_serial_numbers* which are active at a reader within a time period. The concept of the *tag group* emerges in this feature and members of the group are used to derive the other features. Thus, *group_number* is a count of the unique *tag_serial_number's* which are identified at a reader within a time period. The *group_first_tag* is the first tag_serial_number, whereas the *group_last_tag* is the last tag to be identified.

Table 5 - Features of a M:1 association

Feature	Meaning
group_read	The tag serial numbers of tags that appear within a time window at a reader.
group_number	Number of tag serial numbers found in a *tag group*.
group_first_tag	First tag in the group to be *read* at the reader.
group_last_tag	Last tag in the group to be *read* at the reader.
difference_group_time	A general category of features which could be derived by comparison of various features of tags in the group.

As a M:1 association is made up of many 1:1 associations, in a time period, the 1:1 features could be instantiated with values to compare tags within the tag group to each other. For each tag to be compared, its 1:1 features are extracted and measured against those of its fellow group members. This could indicate similarities or differences in how tags have interacted with readers in a system.

6.4.4 MANY-TO-MANY FEATURES

Up to this point, most features have been specific features which can be constructed from the individual 1:1 data records and associations. As the concept of a M:M association subsumes all possible associations in an RFID system, likewise, the features of a M:M association subsume all previous features. Consequently, to avoid

repeating unnecessary information, Table 6 shows a single feature in a single general category of features.

Table 6 - Features of a M:M association

Feature	Meaning
difference_group_reader	A general category of features which could be derived by comparison of tags in the system for all underlying association features.

Table 6 shows a single feature, *difference_group_reader*, which represents a general category of features, which are constructed when many tags are active across many readers or in sequence, that use anti-collision protocols i.e. using all previously constructed features. A tag could be identified at a reader whilst other tags are also being identified there, and then moved onwards to be identified at another reader that is also engaged in identifying tags. In this way, features can be derived on tag groups at each reader and these tag groups can be compared to each other.

To construct features within this general category of features, the requirement is that the underlying associations, and hence features, exist in the system.

To summarise this section, the feature partition contains the features which characterise the *associations* between tags and readers. From the tag and reader perspective, elementary data features are constructed. The increasing complexity of associations, achieved through the addition of more varying tag and reader associations means that more complex features can be constructed from the perspective of a database. These features characterise the underlying associations at the RFID layer, and hence it has been assumed, would exist at layers below this layer. This suggests that the ability to construct new features depends on the underlying associations across the layers of the system, and ultimately, the inclusion of different combinations of tags and readers as the building blocks of the system. In essence, this model has added context to the data view of the RFID domain model.

6.5 SUMMARY

This chapter has described the standard operations for components which exist within the *standard operating partition* from the reference model, and has represented these in a domain model. To constitute what is contained in minor

partitions, and across the layers, it was necessary to use a number of methods. When these were applied to the structure of the reference model, they were systematised for RFID. The resulting domain model is constituted by a logical and data view of the system, and its primary value is in forming a basis for understanding the interrelationships in systems, in addition to a controlled vocabulary for concept identification, ahead of analysing security 'whole of system.'

It is a general principle expounded in this thesis, that one should start with a robust view of 'the system' in deducing what is practicable for security. The approach taken in this chapter, using a variety of analysis methods, from different sources, but integrated through the model, has resulted in a view of the 'the system' and its elements.

Chapter 7

The Problem Partition

This chapter is based on a publication that appeared in IEEE Pervasive Computing, Mobile, and Ubiquitous Systems, Volume 8, Number 4, October-December 2009 (Mirowski et al. 2009b)

7.1 INTRODUCTION

In this chapter a 'whole of system' approach is illustrated within the *problem partition* of the reference model when threats are analysed over system layers. As the *standard operating partition* is layered, and attacks target the elements of these layers, analysing the problem partition considers attacks across the same layers. As these two partitions are aligned to each other, and share common layers, it is possible to compare where in systems it is more effective to address attacks. This next phase in 'whole of system' analysis, facilitated by the reference model, is depicted in Figure 31.

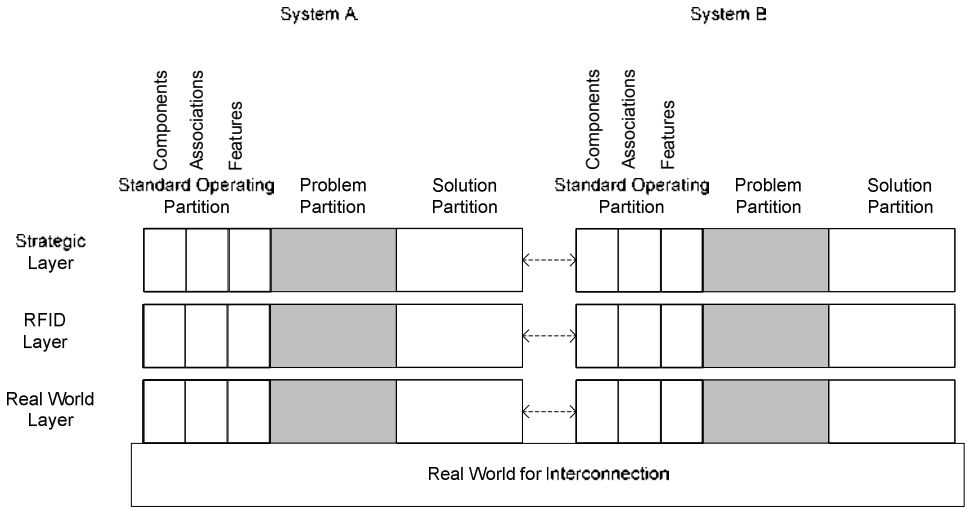

Figure 31 - Analysis of the problem partition
This chapter illustrates how a systematic threat analysis over the layers assists in understanding where attacks in a system occur in addition to identifying more effective locations for solutions.

Recall from Chapter 3 that previous work has analysed individual attacks but usually out of a system context. These range from informal assessments (Juels 2006) to more formal threat analysis techniques (Thompson et al. 2006). Work by Spiekermann and Ziekow (2005) illustrated that threats in RFID systems can be linked together, thereby showing how system goals can be invalidated by an attacker. However, when considering these examples, they do not appear to give explicit consideration to a system's context, and hence, have not established whether attacks are feasible for various system types.

Conversely, working from a system representation, some previous work has achieved some systematisation of threats (Rotter 2008; Mitrokotsa et al. 2010). When

considering these examples, however, threats have appeared localised to system layers or a few parts of a system, which does not sufficiently reflect the feasibility of attacks. Some threats may indeed be constrained to individual layers, but as RFID systems are hierarchical with related layers, attacks have interrelationships which should be considered. The Texas Instruments Digital Signal Transponder (TI-DST) example from Chapter 2 showed that in this system, an attack involved a number of steps – reverse engineering of tags, reprogramming of obtained data, replaying of data, and physical interaction via simulators at a reader – in order for attacks to be enacted. As the RFID system consists of synergistic effects and interrelationships, across layers, consideration should be given to how these influence attack implementation.

To this end, in the problem partition here, *attack sequences* have been modelled over the system layers to develop *An RFID Attacker Behaviour Taxonomy*. This focuses on the hardware layer components that are generic to most RFID systems, such as serial-number-only tags, whilst remaining generic across specific technological implementations like the Electronic Product Code (EPC) system. As most attacks originate in the hardware layer and propagate into the higher layers, using the fact that these systems are hierarchical (see Chapter 2) the taxonomy models how attacks occur in a system, not where the system is physically located. System context comes from this partition's alignment with the standard operating partition.

7.2 AN RFID ATTACKER BEHAVIOUR TAXONOMY

In this section the concept of *An RFID Attacker Behaviour Taxonomy* is introduced to analyse the problem partition 'whole of system'. Attack trees (Schneier 1999, 2004), are used to build the taxonomy to structure the attacks. Attack tree nodes form the taxonomic units: the root node represents the attack goal (the attacker's motivation or incentive for targeting a system), and sub-nodes represent the sub-goals the attacker must achieve to attain that goal. The interconnection between nodes constitutes a sequence that describes an attacker's behaviour in a system. Depending on where they occur in the system determines where they have been modelled at the various layers.

As a basis for thinking about an attacker's motivation when structuring attack sequences over layers, the classification of RFID systems by their information goals (Hassan and Chatterjee 2006) - typically, *authorisation* and *monitoring* – is used. Authorisation systems replace the more traditional approaches of granting a physical entity access to a particular zone, whereas monitoring systems establish a physical entity's location in that zone. Although their information goals differ, the underlying hardware is identical for both types of systems; consequently, attacks can be the same. However, as attacker behaviour invalidates each subsystem's information goals differently, a 'whole of system' approach to security considers these goals individually but models these at the strategic layer.

With these system types in mind, an attacker's behaviour invalidates an RFID system's information goals through attack sequences. As an RFID system consists of elements from the standard operating partition, attacking a system involves attacks against these elements. Tag cloning (see Chapter 2) is one type of attack which has been considered, in addition to other types of attacks. Also, some attack sequences do not involve tag cloning, but still achieve the same attack goal. It is probable that new attacks would eventuate as developments in the field occur, and these attacks could form new nodes. However, fundamentally, the core structures of these attack trees, at the higher layers, would be unlikely to change, and hence, attacker behaviour should remain fundamentally centred on a system's information goals.

Attacks propagating through the layers take on different forms. In Chapter 2 the forms of cloning were presented as an example of this concept. Modelling attack sequences across the layers means remaining constrained by the standard operating partition and its elements at respective layers. For example, in the forms of tag cloning, requires access to a physical tag and would therefore involve attacking the real world layer as well as the RFID layer. Thus, as the organisation of the layers is known, there is a distinct organisation in the way attacks can be sequenced together.

Consequently, the taxonomy represents RFID attacker behaviour in the context of each RFID subsystem's information goals. As the goals define these subsystems, each attack tree's goal is the invalidation of an information goal. Two attack trees are presented - one for authorisation systems and the other for monitoring systems - to ensure that the taxonomy captures the depth of RFID system threats. Although the

attack trees attempt to be as comprehensive as possible, specific system implementation details were omitted, so that the attack trees are generic and scalable. The major attacks were modelled; as new ones emerge, other security practitioners can add branches to these initial attack trees.

Finally, as a way of demonstrating the specific adaptation of the generic taxonomy, an actual system is used as a recurring example. RFID system security is important as some system owners, usually companies, use the information gleaned from this technology to make decisions about high-value entities. One such example stems back to 2004, when the US Food and Drug Administration recommended RFID for product authentication in an effort to eliminate counterfeit pharmaceuticals. Consequently, the drug maker GlaxoSmithKline began placing high frequency, 13.56-MHz tags encoded with an Electronic Product Code (EPC), an industry-standard tag serial number, on Trizivir bottles (O'Connor 2006) to authenticate the drug, which is used to treat patients diagnosed with the human immunodeficiency virus (HIV). The RFID system's ability to determine if a bottle of Trizivir is counterfeit - so that it does not pose a risk to patient health - ultimately depends on overall RFID system security.

Although the standard operating partition has not been explicitly depicted in the diagrams, at this point in the thesis, it is sufficient to discuss these concepts during attack descriptions. Later in Chapter 10, a much larger pharmaceutical supply chain example will be used to illustrate how all layers and partitions of the model can be completely engaged, but for the time being, this chapter limits the model's depictions to the problem partition.

Thus, when considering this example, it is evident that the association between attacks and the system should be established in a structured manner before appropriate security can be formulated.

7.2.1 AUTHORISATION SYSTEM ATTACKER BEHAVIOUR

Figure 32 depicts attacker behaviour in a generic RFID authorisation system, for example, introducing counterfeit pharmaceuticals into an RFID-enabled supply

chain. The attacker's goal, modelled here, is to give an unauthorised physical entity access to the system. [3]

This attack goal has three sub-goals. The attacker starts by obtaining an authorised tag and then attaches it to the unauthorised physical entity that he or she wants to introduce into the system. The final step is for the reader to be able to read the authorised tag attached to the unauthorised physical entity. The reader authorises the physical entity based on its tag ownership and grants the unauthorised physical entity access to the system. The process of carrying out this attack goal involves attacks across various layers in the system.

The attacker's primary task here is to obtain an authorised tag - either an original tag authorised in the system or an original tag's duplicate, called a *clone tag*. Such attacks work when the system authorises a tag solely on its serial number. It can be seen that most work in these attacks occurs *out of band* i.e. not in range of an authorised reader. It is only when these attacks are *in band*, and thus, in range of an authorised reader, that data is produced which will propagate to the higher system layers.

[3] Since the IEEE publication of this diagram, the attack tree now depicts the node *Obtain Tag #* as occurring before *Replay #* node. In both diagrams, layers have also been made explicit.

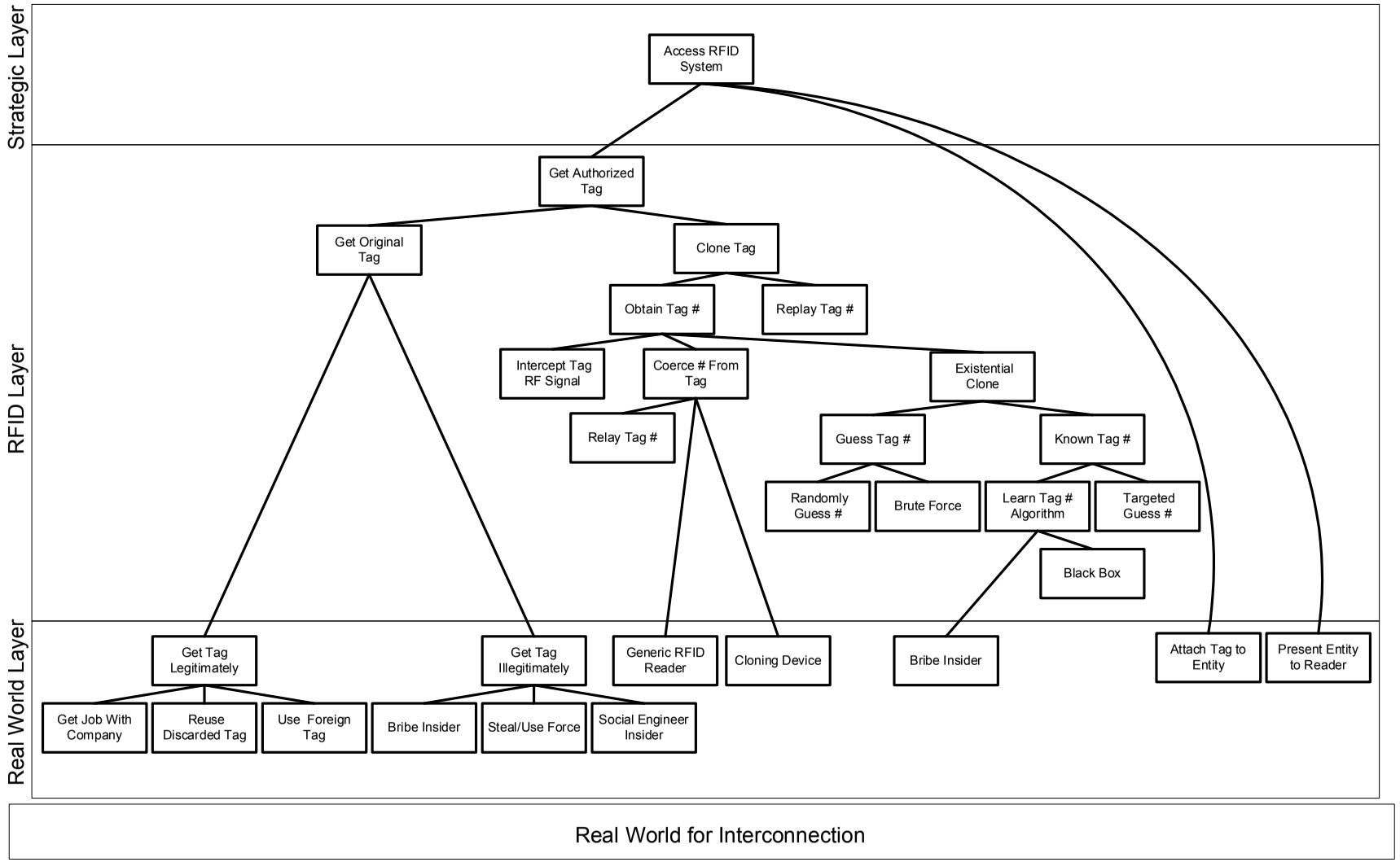

Figure 32 - RFID authorisation system attack tree

An attacker can interfere with an authorisation system by attaching an authorised tag serial number to an unauthorised physical entity. As the reader authorises the physical entity based on tag ownership, it grants the unauthorised physical entity access to the system.

7.2.1.1 ORIGINAL TAG

An attacker can obtain an original tag either legitimately or illegitimately depending on their intent. Some legitimate means of obtaining an original tag would initially not break any laws. A legitimate method would be to retrieve discarded tags from the company's waste, or introduce a tag from a foreign RFID system that has allocated tags from the same serial number space. These attacks can be implemented at the real world layer as they involve direct manipulation of an existing physical tag.

Getting a job with the target company to gain physical access to its tag supply could be legitimate or illegitimate. If the original tag is a tag which was assigned to the attacker as an employee then this is a legitimate attack, whereas if attaining employment was then used to steal tags, the node should be modelled in the illegitimate branch.

Illegitimate means of obtaining an original tag include bribing a company insider into supplying a tag, impersonating a company insider and then stealing an original tag, or exploiting *slap-and-ship* systems. In slap-and-ship systems, tags are attached to physical entities just prior to shipping in order for the company to meet the RFID mandates of downstream companies (Michael and McCathie 2005). Usually there is minimal RFID infrastructure to monitor tags or tagged entities at this stage, thus, a malicious insider might supply an attacker with tags primed to compromise the system, and the system may not realise that tags are missing. Alternatively, attackers might steal or use force to obtain a supply of original tags, removing them from a legitimate product and then using them on their own unauthorised physical entities.

This branch of attacks is feasible when the RFID system authorises a physical entity based solely on a tag's stored number. The only limitation to the widespread use of these attacks is the number of original tags available. Attacks like this are easy when physical security is weak.

7.2.1.2 CLONE TAG

Systems that authorise physical entities based on a tag's serial number are also vulnerable to tag cloning. The attacker obtains an authorised tag's serial number and then replays it back to a reader, usually by writing the serial number to a reprogrammable tag or using a device that can simulate the tag's radio signals.

These are usually engaged more at the RFID layer as they are relying on things like radio signals, anti-collision, and specific RFID system information for them to be implemented.

An attacker can acquire an authorised tag serial number in several ways. An existential cloning attack, for example, lets attackers guess an authorised tag's serial number in the target system. EPC tags are vulnerable to such attacks (Juels 2005), as is the human-implantable VeriChip (Halamka et al. 2006) which hospitals use to identify medical patients. Randomly guessing tag serial numbers using the tag's number space or sequentially working through the number space - called a *brute force attack* - might also provide an authorised tag's serial number.

Two examples highlight the weak security assumptions that so called *uncloneable* tags - tags protected by cryptography - rely on, showing their vulnerability to brute force attacks. The 40-bit secret key of a Texas Instruments Digital Signal Transponder (DST) tag was recovered in less than one hour (Bono et al. 2005) and a MiFare Classic smart card secret key was retrieved within a few seconds (Garcia et al. 2008; Nohl et al. 2008; Courtois 2009). More details of these attacks were discussed in Chapter 2. When considering these examples, it illustrates how the disparity between onboard tag capabilities and readily accessible computer hardware is a problem for effective protection against such attacks at the tag.

Moreover, attackers might have domain knowledge that assists them in determining an authorised tag serial number. They might pay a company insider to reveal how the number space is populated with tag serial numbers, or they could perform a *black-box attack*, whereby they obtain a block of tag serial numbers to determine their commonality. Sometimes it is not necessary to have any advanced domain knowledge, just access to an existing tag - for example, it was observed that some proximity cards print their serial number right on the card in case a security officer has to physically inspect a tag. Likewise, an EPC number's barcode representation is sometimes printed on the tag's casing.

Another way to obtain an authorised tag serial number is by coercing a tag into revealing its serial number. Attackers might use an off-the-shelf RFID reader to read a tag's serial number directly. Alternatively, they can use a tag-cloning device to

capture tag radio signals and thus tag serial numbers, which they then store for later use. Westhues (2005) reports that for less than US$100, attackers can build an electronic device to capture and replay tag radio signals. This is substantially cheaper than purchasing a legitimate reader to achieve the same effect.

Although tag cloning is typically the passive act of recording a tag's serial number, tag relaying and tag radio frequency signal interception are active attacks that operate in real time to achieve the same effect. Attackers can intercept tag radio signals while an authorised tag and reader are communicating, then reroute the signals so that they appear to be emanating from the unauthorised tag and physical entity (Hancke 2006). These attacks demonstrate how tag-to-reader communications are not safe even at short distances.

Thus, systematisation for these attacks reveals ways in which the attacker can sequence individual attacks over layers to invalidate a system's goals.

7.2.2 MONITORING SYSTEM ATTACKER BEHAVIOUR

Attackers might also try to prevent monitoring systems from producing information used to track the location of physical entities - for example, to enable the removal of products from a supply chain without being detected. Figure 33 illustrates attacker behaviour in a generic RFID monitoring system. As the RFID layer is responsible for bridging the real world layer and the strategic layer, it is the most vulnerable layer to these sorts of attacks.

Three ways which an attacker can use to achieve the attack goal of preventing an RFID monitoring system from working correctly were identified: prevent a tag from identifying itself to a reader; prevent a reader from identifying a tag; or prevent a database located in the RFID middleware from associating a tag with an authorised physical entity. Each of these methods is now discussed in more detail.

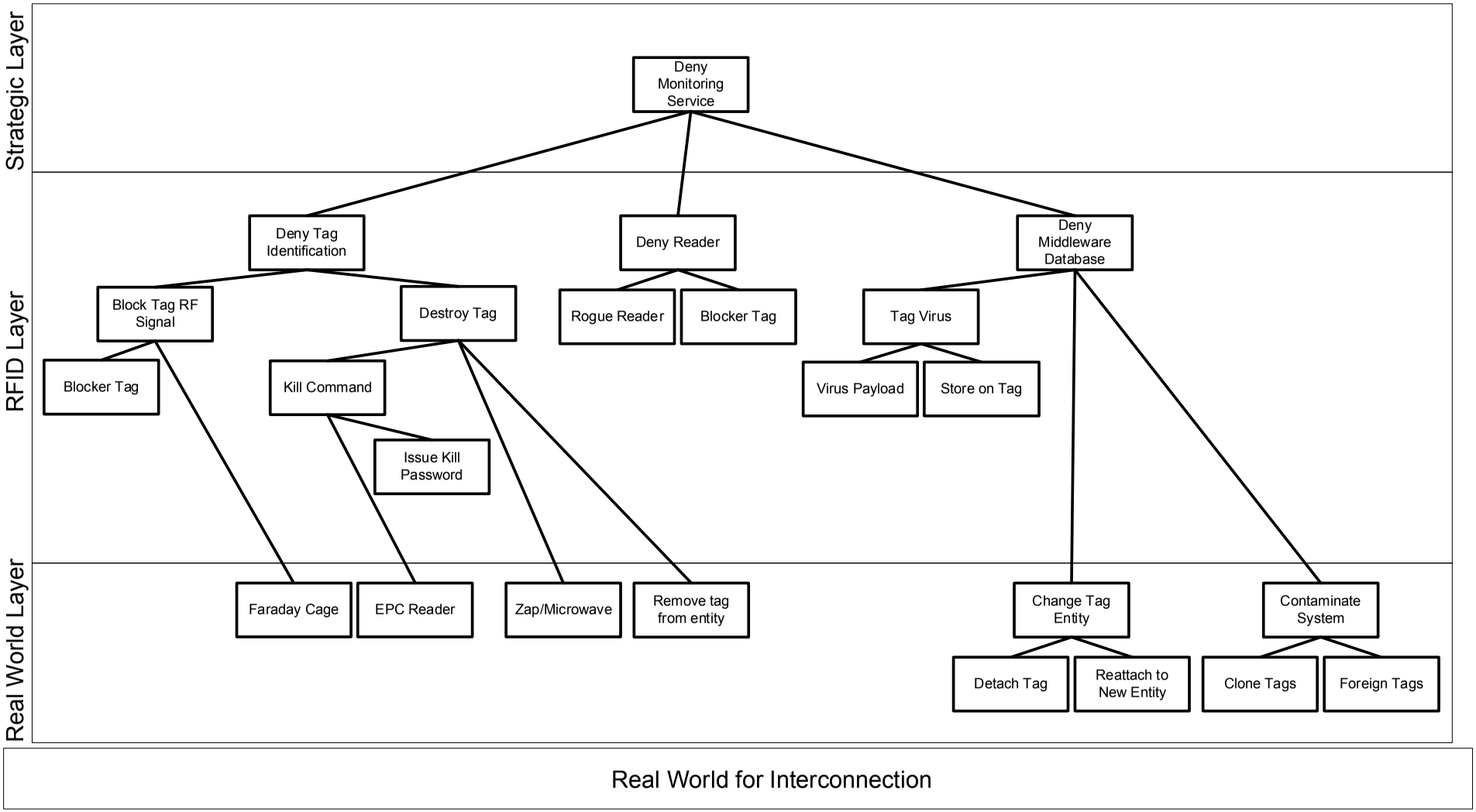

Figure 33 - RFID monitoring system attack tree

Attackers can interfere with the monitoring system by preventing a tag from identifying itself to a reader, preventing a reader from identifying a tag, or preventing a database located in the RFID middleware from associating a tag with an authorised physical entity.

7.2.2.1 DENY TAG IDENTIFICATION

When attempting to deny tag identification, an attacker can exploit a *blocker tag* (Juels et al. 2003) which prevents readers from resolving individual tag serial numbers. Conversely, attackers can use a *Faraday enclosure* to block an individual tag's radio frequency signal. A Faraday enclosure works by shielding a tag's signals such that they cannot be detected by a reader (Hashemi 2009) and such an approach could be used by petty thieves in bypassing a store's Electronic Article Surveillance (EAS) systems.

If attackers have access to a tag, they might also modify it physically to prevent tag identification. The *Kill command* is a built-in operation of EPC tags and, when issued, destroys a tag's ability to be identified (Glover and Bhatt 2006). Although access to the command is password protected, attackers have used power-analysis attacks to determine the secret kill password for UHF Class-One Generation-One EPC tags (Oren and Shamir 2007) which means that attackers can destroy tags if they can get close enough. The household microwave oven is another effective way of destroying a tag; the heat damages a tag's sensitive electronic components. Collins (2006) reported that attackers can use a modified camera and flash just as effectively. In addition, the RFID Zapper device can emit an electronic pulse strong enough to deactivate tag electronics. However, the simplest method of attack is removing a tag from its associated physical entity. Clearly, this branch of attacks invalidates the security assumptions that tags always respond to readers and that tags and physical entities are inseparable.

7.2.2.2 DENY READER

As the reader is responsible for associating a tag with a physical entity, denying reader service would mean that the system cannot monitor tags and therefore physical entities. Due to the anti-collision algorithms that most readers use, a tag typically communicates with only a single reader at a time (although some standards such as EPC Class-One Generation-Two (EPCglobal 2005) enable tags to interact with multiple readers simultaneously using sessions). An attacker might exploit this and introduce his or her own reader to keep a tag in a temporary busy state, thus delaying or preventing it from communicating with legitimate readers. Similarly, a Blocker Tag (Juels et al. 2003) keeps an authorised reader busy trying to schedule

tags to respond. Of course system context is important here, as it is reported that the Blocker Tag is targeted at systems which use Singulation Tree Walking protocols to identify tags. As the tag cannot identify itself, these attacks prevent the reader from associating a tag with a physical entity.

7.2.2.3 DENY MIDDLEWARE DATABASE

The last branch of attacks denies a monitoring service access to the database and its stored event information. As a tag generates this information when a reader passes by, attacks against these components result in attacks against the middleware database.

Rieback et al. (2006) have proposed the tag virus as a way of introducing malicious information into a database. Attackers might construct a malicious data payload, such as a computer virus stored on a tag's memory, to destroy critical data. The attacker can transmit the virus to the reader during a tag read operation and then store it inside the database. When the database executes the virus, it performs a malicious operation, such as deleting data records. An administrator might need to take the database offline for repair, delaying new incoming tag information.

News sources have reported cases in which attackers have successfully encoded a malicious payload onto a tag. For example, one attacker encoded a payload in a JPEG-2000 image file on an electronic passport (epassport), which caused the reader to crash (Zetter 2007). Specifically, the attacker embedded a buffer-overrun exploit in the JPEG-2000 file on the cloned chip containing the epassport photo (several airports around the world use RFID readers vulnerable to this type of attack). Although this attack targets readers, it proves that attackers can target tags and then execute their own instructions on the vulnerable computer and spread them to the middleware database. Clearly, these attacks take advantage of the hierarchical nature of RFID systems (see Chapter 2).

Some of the attacks which appear in Figure 32 have been appended to this branch of attacks – namely the tag cloning attacks.

Attackers can also deny a middleware database or a monitoring service by changing the association between a tag and a physical entity - for example, by detaching a tag from a physical entity and attaching that tag to a completely different physical entity.

Dissociating a tag from its physical entity, also known as a *change-of-tag ownership* attack (Mirowski and Hartnett 2007) invalidates the information in the database. When an attacker swaps a tag from pallet A for a tag on pallet B, for instance, the database will record these events against the tag from pallet B when pallet A appears at a reader. Likewise, when the reader locates pallet B, pallet A will be in its location.

Finally, attackers might introduce clone or foreign tags into the RFID monitoring system, which modifies the association between tags and physical entities. Introducing foreign tags – tags which are active in a different RFID system – into a system where the same tag serial numbers are already active, is a form of *cross contamination*. Cross-contamination allows foreign tags to achieve the same privileges as authorised tags (Heydt-Benjamin et al. 2006). Any of the other ways clones can be introduced (see Figure 32) could also contribute to this attack goal. As a clone is a duplicate of a tag serial number that is already active in a system, these attacks record information from several physical entities in the database against the same tag. The system does not validate this information, so invalid information propagates up to the higher RFID layers.

7.3 DISCUSSION

While the taxonomy was developed to analyse the problem partition of systems 'whole of system', this section demonstrates how it can be applied to a theoretical attacker scenario. An attacker, Tyrell Corporation, manufactures a counterfeit version of the Trizivir drug, Nexus-6, and sells it to people diagnosed with HIV. As the association between a tag and a Trizivir bottle indicates an authorised Trizivir drug, the attack goal is to gain access to the supply chain.

The drug accesses the supply chain based on the tag's serial number and its related records stored in the middleware database. Invalidating the authorisation subsystem means invalidating the RFID information so that it reports the counterfeit drug as the authorised drug. Therefore, the authorisation attack tree in Figure 32 is used to analyse the attack in this system context:

- Tyrell Corporation purchases a generic reader from an RFID manufacturer.
- Tyrell scans a tag in the Trizivir supply chain by purchasing a bottle of Trizivir; alternatively, it scans a drug delivery truck as it passes by the reader.
- Next, Tyrell obtains the tag identifier from a tagged Trizivir bottle and encodes it on a reprogrammable tag that it will replay later (this constitutes a clone tag). As the system authenticates the tag, and hence the entity, based on serial number, Tyrell Corporation has effectively obtained an authorised tag.
- It then attaches the clone tags to its bottles of Nexus-6 and introduces them into the supply chain, perhaps with a malicious insider's assistance.
- Finally, the Trizivir system reads the tags attached to the Nexus-6 bottles. The clone tag serial numbers correspond to authorised serial numbers in the middleware database. As the tag is a surrogate for the drug, the drug is deemed authentic, giving Tyrell Corporation access to the system. The counterfeit Nexus-6 drugs are now authorised and allowed into the system in place of the authentic Trizivir.

Understanding this attack sequence is useful when deploying countermeasures as it helps to identify the types of attacks to which the RFID system is vulnerable. In this scenario, the system is vulnerable to a tag-cloning attack as it is a serial-number-only system. Attackers can duplicate tags if they obtain the authorised tag's serial numbers.

The RFID system's security assumptions can also be identified. In this scenario, the system assumes the physical entity to which a tag is attached is not counterfeit. The system makes no attempt to authenticate the physical entity, simply choosing to believe the tag's authenticity. The system also assumes the tag's serial numbers are all unique and not in use by multiple tags. After a tag leaves the system, it assumes that the tag will not be reintroduced into the system for another drug to use. Furthermore, the middleware database assumes that the event-oriented information that the reader produced when reading the tags accurately reflects what has occurred in the real world. (In Chapter 10, these concepts, which fit the standard operating

partition, will be made explicit in the specific example of a larger pharmaceutical supply chain case study).

This knowledge can be used to determine where to place system security countermeasures. In this scenario, the key vulnerability is that the drug is authorised based on the tag's serial number. Attackers can replace the original drug with a counterfeit drug without modifying the legitimate tag as nothing links the tag to the physical drug apart from information residing in the middleware database. To improve security, the manufacturer could, for example, choose to include information on the tag to link it to the drug itself.

As the attack trees indicate, entities may also be vulnerable to change-of-tag-ownership. Therefore, a more effective solution may be to implement an intrusion detection system. In RFID, these solutions analyse the RFID data entities produce when their tags interact at a reader. Mirowski and Hartnett (2007) report that a simple intrusion detection, which computes statistics on entity behaviour, can indicate when an implausibility has been detected. This solution takes advantage of the RFID data attesting to entity behaviour throughout the system, meaning that even if tags change entities, these attacks could be detected.

In essence, when the attacks in the problem partition are analysed across the layers in this partition, using a systematic method such as attack trees, it is possible to think of RFID security requirements with respect to the system's context.

7.4 SUMMARY

This chapter has illustrated the benefits of applying a 'whole of system' approach to threat analysis via the reference model. It used the reference model's problem partition to this end, and undertook threat analysis using the attack tree method.

Attack trees formed a taxonomy which models attacks at various layers as attack sequences. As these systems are layered, attacks were represented as sequences over the layers for authorisation and monitoring system types. The attacker's attack goal was the invalidation of a system's information goal, while sub-goals represented ways of achieving the attack goal. As an example of analysis of an attack which would benefit from systemisation, tag cloning attacks were placed in the context of

systems, and other attacks at different layers were used to show how these would support tag cloning.

As a general principal of systematisation is advocated using layers in this partition, the use of systematic threat analysis, over layers, means attacks can be aligned against the standard operating partition to capture both generic and specific system properties. Attack trees represent one such systematic threat analysis method. Others may also exist, and would therefore, also be relevant. Systematic threat analysis has been shown to reveal the locations which can be targeted by the attacker to invalidate a system's goals. Thus, the recommendation is that solutions are located where they are more feasible in the system hierarchy.

The knowledge imparted here is that applying a 'whole of system' approach to threat analysis, by considering attacks using a systematic method over system layers, leads to a more effective identification of which threats are feasible in a system context.

Chapter 8

The Solution Partition

This chapter is partially based on a publication presented at the 5th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne, Australia, 2009 (Mirowski et al. 2009c)

This chapter is also partially based on a publication presented at the 10th International Symposium on Pervasive Systems, Algorithms, and Networks, Kaohsiung, Taiwan, 2009 (Mirowski et al. 2009a)

8.1 INTRODUCTION

In this chapter a 'whole of system' approach is illustrated through analysis of the *solution partition* in the reference model. The solution partition is the area where solutions, which can address attacks, are analysed in the model. As the final major partition, this partition largely depends on the organisation of the elements of the *standard operating partition* in determining what solutions are practicable for systems, in addition to the threats in the *problem partition,* in identifying what threats warrant countermeasures.

Consequently, as the partitions are aligned across system layers, this chapter considers how analysing solutions using a 'whole of system' approach, made possible by the reference model, can suggest practicable solutions. The alignment of the partition explored in this chapter, in relation to the other partitions, is illustrated in Figure 34. This illustration emphasizes the need to define a system's context, and the threats in that context, for solutions to be analysed.

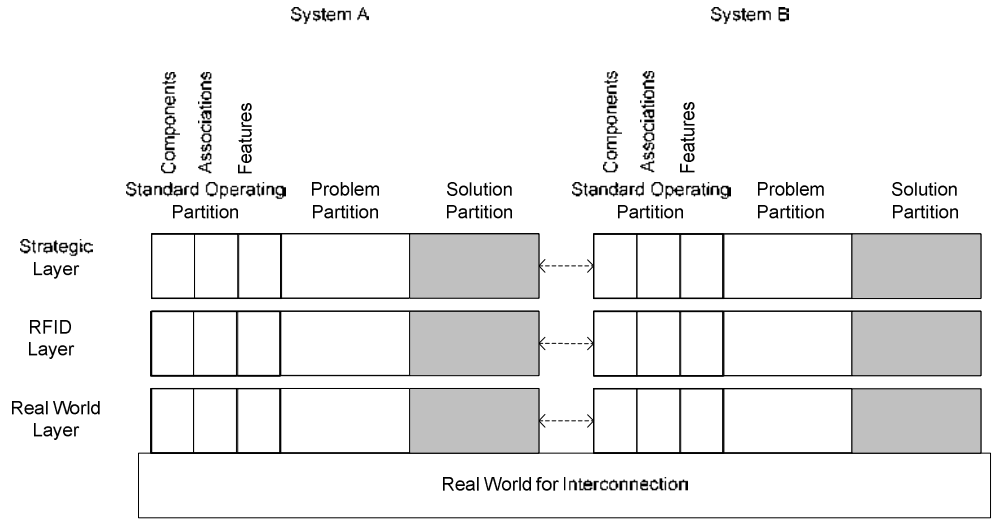

Figure 34 - Analysis of the solution partition
This chapter illustrates how a 'whole of system' approach to security analysis can proceed in the solution partition by introducing a simulation model for this purpose.

To this end, this chapter introduces the concept of an RFID simulator which has been validated for 'whole of system' analysis (see Appendix A). It is based on the domain

model introduced in Chapter 6 and on agent based modelling and simulation (ABMS) concepts (Robinson 2004) and allows systems to be modelled and attacks introduced. The simulator is used to explore the problem of exposing clone tags in an RFID system, to demonstrate this approach to solution analysis. The previously unexplored solution potential of system associations (see Chapter 6) to expose attacks in RFID is modelled. Output data is produced via the simulator and analysed for attacks. Various associations and features which influence attack exposure are discussed. This process represents the use of elements, integrated across a system, for solution analysis. In exploring these factors in simulation, the benefits of approaching solution analysis through a 'whole of system' approach are made apparent.

8.2 EXPOSING ATTACKS IN SYSTEMS

In order to understand a solution, one needs to consider the threats and the system context, in addition to what parts of the system the solution is reliant upon. The need for this approach to analysis is evident when considering the class of security solutions called *Intrusion Detection*. Intrusion Detection can be separated into two sub-classes: plausibility checks, and anomaly detection. These solutions, in general, rely on elements across system layers in order to identify attacks. In developing these types of solutions, or deploying them, consideration must be given to the elements in various parts of the system. Thus, briefly reviewing these will highlight the need to take a 'whole of system' approach to solution analysis.

Plausibility checks formalise what constitutes an attack, using such things as rules or models, applying these to identify attacks in system data. Koh et al. (2003) proposed a track and trace system whereby an audit trail is established containing a history of all the procedures a product has undergone at each stage in a supply chain. In this approach, a user manually verifies whether a process is valid. Conversely, an automated approach was proposed by Illic et al. (2009) relying on rules to check basic supply chain conditions as valid. These rules include checks on: *velocity*; *dwell-time*; *lifecycle; pair-wise shipping and receiving*; and *transition probability*. In order for these solutions to be analysed they need to be placed into the context of an actual system where attacks are detected via these means.

Conversely, *anomaly* approaches automatically formulate a profile of normal activity and use this to determine what constitutes an attack. In the area of clone tag detection several approaches exist. Mirowski and Hartnett (2007) proposed a system that uses statistical anomaly detection to identify the behaviour of clone tags. Lehtonen et al. (2007a) proposed two approaches for identifying anomalies in complete RFID traces which could be due to clone tags: a supply chain model (SCM) and a Hidden Markov Model (HMM). Similarly, Lehtonen et al. (2009) applied intrusion detection approaches to single events instead of complete data traces. A stochastic supply chain model was used to illustrate different real world problems in 'location based' product authentication. Two probabilities for genuine tags were calculated: *location-transition* probability and *time-transition* probability. Although profiles are learnt automatically, unless consideration is given to a systems design, then the right data may not be instantiated such that the solution exposes attacks.

When considering the above examples, it seems likely that careful consideration must be given in order for attacks to be exposed by these solutions. To analyse these solutions, the influences of the system need to be considered, and this would mean placing these solutions into a system. The next section shows how applying a 'whole of system' approach can analyse such solutions using a simulation model.

8.3 ANALYSING A SOLUTION 'WHOLE OF SYSTEM'

This section illustrates how a 'whole of system' approach in the solution partition occurs. It uses the problem of exposing clone tags in RFID data to consider how a system's associations influence the possibility of detection. To facilitate this investigation, an RFID simulator is introduced, and the results generated using it, are presented as the basis for analysis of the solution's effectiveness. In exploring this example, the approach towards solution analysis in this partition is illustrated.

The reference model promotes the concept that solutions can be dealt with at various layers. The segment of a layer selected here, a combination of associations and features, is where it may first be apparent that cloning attacks could have arisen in RFID data. The exposing of clones here relies upon the occurrence of multiple tags which contain

the same serial number in the system. When such tags are prevalent, the method identifies their occurrence through the implausibilities they produce in the constructed features and associations. Such attacks would be suitable for detecting tag cloning and pseudo-cloning (see Chapter 2) as in these attacks usually the original tag is also active in the system. Therefore, rather than explore the same problem across all parts of the model, the reference model suggests this is perhaps the first place cloning can be addressed in data.

8.3.1 RFID SIMULATOR

The software based simulator for RFID systems is based on the domain model in Chapter 6 and has been validated for preliminary analysis tasks. It fills a gap in RFID security research whereby researchers would like to investigate attacks in actual systems; however, these actual systems, and systems with attacks in them, are not available. In addition, sometimes it is also not feasible to begin research in an actual system, as the researcher can have less control over the behaviour of components or the speed in which systems elicit information useful to investigations. This section provides a brief overview of the simulator and more information can be found in Appendix A.

The simulator uses *agent based modelling and simulation* (ABMS) principles (Macal and North 2005; Korth 2006) and was implemented in software using an existing ABMS toolkit called MASON (Balan et al. 2003; Luke et al. 2004). In the simulator, a user builds up an RFID system using the controlled vocabulary from the domain model. To enable this, an application programming interface (API), (a set of commands related to domain concepts), is engaged to compile a script. The script indicates how the system should be designed, and the starting conditions for all components. Execution of a scenario results in an animation being run on screen, which enables the user to visualise the system output as if it were actually built. Another output is data which attests to tag and reader interactions. Completion of a simulated scenario results in information useful to the researcher, which would be similar to that produced by an actual system.

In this chapter, the simulator is used to encode a theoretical cloning scenario, and the results are presented around the discussion on the 'whole of system' approach to solution analysis through tag and reader associations as a solution.

8.3.2 SIMULATED SCENARIO: OVERVIEW AND SETTINGS OF TAG CLONING ATTACK

The simulation facilitates a solution to be examined across layers for a scenario. To this end, a tag cloning scenario is introduced. Its settings in the simulator are described for the intended goal of examining how clone tags are expressed in output data. In the modelled system containing clone tags, clone tags interact with readers at different times, and RFID data is produced. Complicating the scenario is the presence of non-clone *original* tags which bear the same tag serial number. The results of simulation are presented, and a discussion on solution feasibility (how the system influences attack exposure through several system layers) is analysed.

The settings for the simulated scenario are now explained. The simulated system can be defined as a many-to-many (M:M) configuration as any type of association could emerge (see Chapter 6). All of the associations take place in a single *zone* which measures 50 centimetres (cm) cubed. In this environment three *readers* are located in sequence: reader_1 is positioned at 10 cm from the origin (0, 0, 0) in the zone, reader_2 is positioned at 20 cm, and reader_3 is positioned at 30 cm. These readers are located over the route which tags were instructed to follow. There are two types of tags: originals and clones. The originals, of which there is one, has the tag serial number: *tag_1_(tag_1)*. The clones, of which there are two, have the tag serial numbers: *tag_2_(tag_1)* and *tag_3_(tag_1)*. The name in brackets signifies a tag's public name which is the serial number that real systems obtain when a tag is read, whereas the name external to the brackets is a tag's private name used to identify it within the simulation. It can be seen that the clones are duplicating tag_1. In real systems this differentiation in serial numbers is not possible, hence, the need for intrusion detection systems.

The components have been instructed to move within the zone and their interactions lead to output data. Tag_1_(tag_1) moves from position 0 cm to position 50 cm at a

speed of 5 cm per second. Tag_2_(tag_1) moves from position 0 cm to position 50 cm at a speed of 1 cm per second. Tag_3_(tag_1) moves from position 50 cm to position 0 cm at a speed of 5 cm per second. The tags move along the route covered by the read range of the three readers. As some of these components move in different directions, different speeds, and at different times, it results in different associations, and hence values, arising in the system. Consequently, some of these behaviours have been identified as implausible, and linked to the existence of clone tags. The use of public and private names made it possible to differentiate these records by the user in addition to visualising the animation.

The results which were generated in the simulated scenario are now presented in figures and a table. The figures (Figure 35 to Figure 40) represent the simulated scenario animation at discrete time steps. They illustrate the various components on screen such as tags and readers. Conversely, the table (Table 7) contains the raw data values for the simple and constructed features which were first proposed and described in Chapter 6, and adapted for use in this scenario. Subsequent sections of this chapter will provide further explanation of these simulation results.

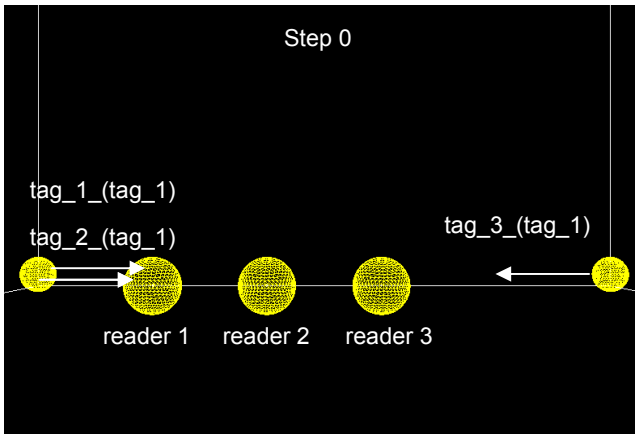

Figure 35 – Simulation Step 0
No tags have interacted with readers, thus, the system has no record of system activity.

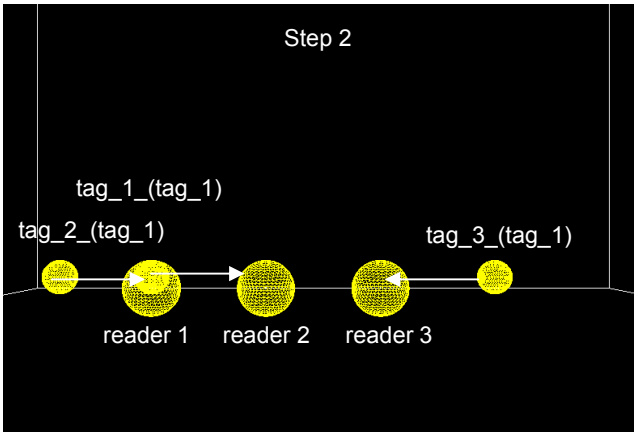

Figure 36 – Simulation Step 2
Tag_1_(tag_1) interacting with reader_1 as it travels to its destination.

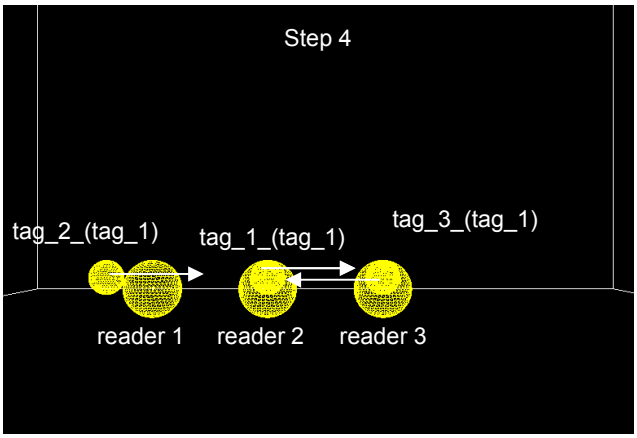

Figure 37 – Simulation Step 4
Tag_1_(tag_1) interacting with reader_2, and tag_3_(tag_1) with reader_3 simultaneously. The readers will report tag_1 as having been read.

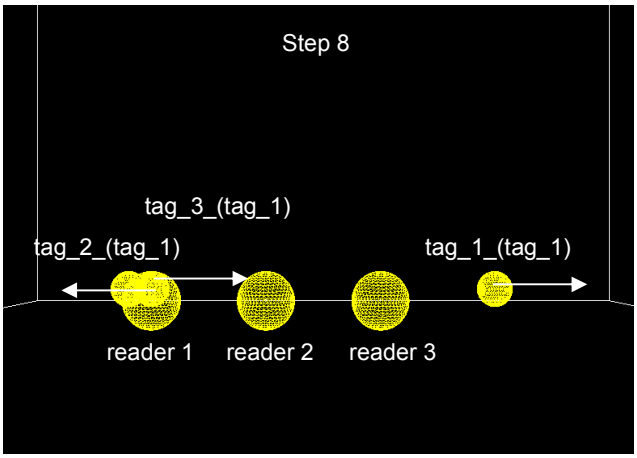

Figure 38 – Simulation Step 8
Tag_2_(tag_1) and tag_3_(tag_1) interacting with reader_1 much faster than normal for a single tag, and also after tag_1_(tag_1) has already been seen at reader_2 and reader_3.

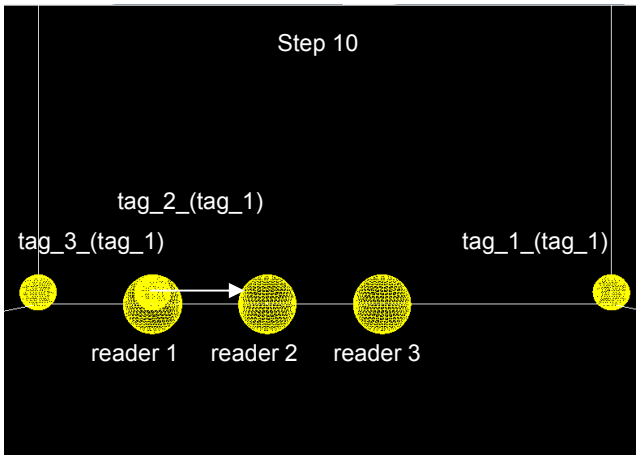

Figure 39 – Simulation Step 10
Tag_2_(tag_1) interacting with reader_1 still, which contradicts reader_2 and reader_3 as having already seen this serial number.

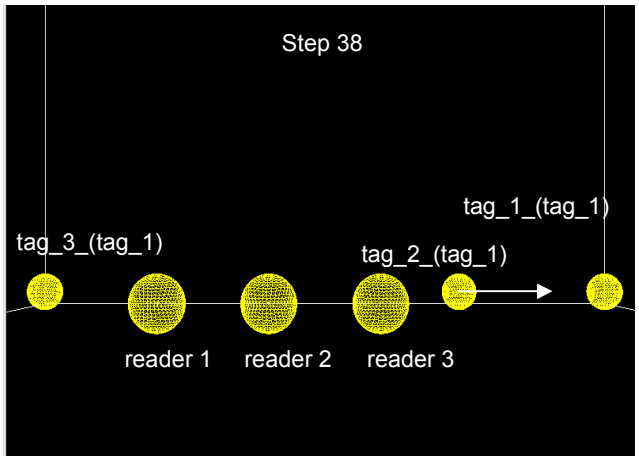

Figure 40 – Simulation Step 38
All tags have finished interacting with readers, the last reader interacted with was reader_3.

Table 7 – RFID output data from the simulated cloning scenario
The constructed features were derived from the associations which arose when components interacted.

session_ number	date	time	reader_serial _number	tag_serial_ number	reader_ operation	total_ number_of_ tag_reads_at_reader	total_ tag _reads	time_since_ last_seen_at_ this_reader	time_ between same_tag_at_different_readers (sec)	previous_ reader_tag_ read_at
1	5/01/2009	4:40:07	reader_1	tag_1_(tag_1)	read	1	1	0	0	null
2	5/01/2009	4:40:08	reader_1	tag_1_(tag_1)	read	2	2	1	0	reader_1
3	5/01/2009	4:40:09	reader_3	tag_3_(tag_1)	read	1	3	0	1	reader_1
4	5/01/2009	4:40:09	reader_2	tag_1_(tag_1)	read	1	4	0	0	reader_3
5	5/01/2009	4:40:10	reader_3	tag_3_(tag_1)	read	2	5	1	1	reader_2
6	5/01/2009	4:40:11	reader_2	tag_3_(tag_1)	read	2	6	2	1	reader_3
7	5/01/2009	4:40:11	reader_3	tag_1_(tag_1)	read	3	7	1	0	reader_2
8	5/01/2009	4:40:12	reader_2	tag_3_(tag_1)	read	3	8	1	1	reader_3
9	5/01/2009	4:40:12	reader_3	tag_1_(tag_1)	read	4	9	1	0	reader_2
10	5/01/2009	4:40:13	reader_1	tag_2_(tag_1)	read	3	10	5	1	reader_3
11	5/01/2009	4:40:13	reader_1	tag_3_(tag_1)	read	4	11	0	1	reader_1
12	5/01/2009	4:40:14	reader_1	tag_2_(tag_1)	read	5	12	1	2	reader_1
13	5/01/2009	4:40:15	reader_1	tag_2_(tag_1)	read	6	13	1	3	reader_1
14	5/01/2009	4:40:16	reader_1	tag_2_(tag_1)	read	7	14	1	4	reader_1
15	5/01/2009	4:40:17	reader_1	tag_2_(tag_1)	read	8	15	1	5	reader_1
16	5/01/2009	4:40:18	reader_1	tag_2_(tag_1)	read	9	16	1	6	reader_1
17	5/01/2009	4:40:24	reader_2	tag_2_(tag_1)	read	4	17	12	6	reader_1
18	5/01/2009	4:40:25	reader_2	tag_2_(tag_1)	read	5	18	1	7	reader_2
19	5/01/2009	4:40:26	reader_2	tag_2_(tag_1)	read	6	19	1	8	reader_2
20	5/01/2009	4:40:27	reader_2	tag_2_(tag_1)	read	7	20	1	9	reader_2
21	5/01/2009	4:40:28	reader_2	tag_2_(tag_1)	read	8	21	1	10	reader_2
22	5/01/2009	4:40:33	reader_3	tag_2_(tag_1)	read	5	22	21	5	reader_2
23	5/01/2009	4:40:34	reader_3	tag_2_(tag_1)	read	6	23	1	6	reader_3
24	5/01/2009	4:40:35	reader_3	tag_2_(tag_1)	read	7	24	1	7	reader_3
25	5/01/2009	4:40:36	reader_3	tag_2_(tag_1)	read	8	25	1	8	reader_3
26	5/01/2009	4:40:37	reader_3	tag_2_(tag_1)	read	9	26	1	9	reader_3
27	5/01/2009	4:40:38	reader_3	tag_2_(tag_1)	read	10	27	1	10	reader_3

Table 7 depicts the features which have been constructed from the associations which arose during the simulation of the scenario. These have been instantiated using the raw data produced when tags and readers interacted which has been illustrated in Figure 35 to Figure 40. The elementary features constructed are: *tag_serial_number*, *reader_serial_number*, *reader_operation*, and *time_stamp* (date and time). For plausibility checks, this data is the source of information about the simulation events which have been visualised in the animation's graphical user interface (GUI). As associations arose between the tags and readers, more complex features were derived. The full range of associations emerged at various stages of the simulation from one-to-one (1:1) associations through to many-to-many (M:M) associations.

To this end, these features and the data in them are now examined in an attempt to identify implausibilities which may expose clones. These implausibilities are due to the various associations permeating through the system layers to the data layer.

8.3.3 SCENARIO ANALYSIS

The first indication of 'whole of system' analysis comes from directly using the simulation model for solution exploration. A benefit of analysing the solution partition through a simulator is that multiple outputs are available. In this case, both animation and raw data of a simulated RFID system can be used to analyse simultaneously. The rest of this section uses the simulator features, and shows how the layers and features arose, to illustrate this approach to solution analysis.

The *total_number_of_tag_reads_at_reader* feature is the number of data records a tag serial number has produced at a reader. Session_number 2 indicates that the original tag, tag_1_(tag_1), has been read at reader_1 twice (that is, in actual systems, at least the original tag serial number). Session_number 3 indicates that tag_3_(tag_1), a clone tag, has been read at reader_3 a single time. This suggests that a clone tag is present as no records were produced at reader_2 which should have been activated in order for a tag to have been read at reader_1 and reader_3 – it does not however indicate which tag is actually a clone. This feature exposes a clone tag when a tag produces an unexpected number of output data records at a reader.

The *total_tag_reads* feature is the total number of data records produced by all tags at all readers. When used in conjunction with the *total_number_of_tag_reads_at_reader* feature, inconsistencies due to the system's sequential ordering of readers are apparent. Session_number 3 indicates when tag_3_(tag_1) is read, overall three reads have occurred in the system; however only one read has occurred at the current reader, which is reader_3. Two reads remain unaccounted for. This suggests that this tag serial number has been active at readers other than reader_3. This can be seen in session_number 1 and session_number 2 which were produced by the original tag, tag_1_(tag_1). Thus, a lack of activity at reader_2 is evidence to suggest a clone is present. These features expose a clone tag when an unexpected number of output data records when compared to each reader in a system are identified.

The *time_since_last_seen_at_this_reader* feature is the time in seconds between successive tag reads at the same reader. There are a number of instances where the delay between a tag having been read in successive times at a reader is 0 seconds. This can be seen in session_number 11. The reason for the very low response is that tag_2_(tag_1) is also being read at this reader, as seen in session_number 10. It can also be seen that this occurs at session_number 3 and session_number 4 for other tags. However, this is as it is the first time the tag serial has appeared at these readers, which would mean other features would be needed to make an accurate identification in these cases. However, in general, there should be a delay between successive reads of the same tag at a reader, on the basis that anti-collision introduces strict scheduling between successive reads of the same tag serial number (this is further explored in Chapter 9). This feature exposes a clone tag when the tag responds to a reader request faster than would be expected for the reader to handle such requests.

The *time_between_same_tag_at_different_readers* feature is the time between a tag's successive reads at different readers. It can be seen that there are a number of instances where the delay between a tag's successive reads at different readers is 1 second or less. It is reasonable to assume, for the simulated system, that it would take at least 1 second to travel between different readers in this simulation given the physical distance between them. Session_number 9 indicates a delay of 0 seconds for the original tag, tag_1_(tag_1), appearing at reader_3. This is due to a clone tag,

tag_3_(tag_1) being read at reader_2 at the same time in session_number 8. Furthermore, session_number 12 indicates a delay of 2 seconds between tag_2_(tag_1), a clone tag, being read at reader_1, and the same tag serial number being read at reader_3, 2 seconds prior in session_number 9. It can also be seen that no data records were produced at reader_2 during the 2 second time period, as the tag serial appeared to have travelled from reader_3 to reader_1. This feature exposes a clone tag when a tag has been read at different readers faster than expected for the physical ordering of readers.

Finally, the *previous_reader_tag_read_at* feature is the reader a tag was previously read at. In this simulation there is a strict ordering of how tags can travel between readers: reader_1, reader_2, reader_3 – or vice versa. Therefore, a tag should travel according to this ordering. It can be seen in session_number 4 that tag_1_(tag_1) is read at reader_2 and the feature indicates it was previously read at reader_3. This is due to session_number 3 having been produced by a clone tag, tag_3_(tag_1). This suggests that the tag is travelling backwards, from reader_3 to reader_2. However, further examination of the data records shows that the tag was previously seen at reader_1 prior to this record which raises questions about how it appeared at reader_3 – clearly there must be another tag in the system using the same tag serial number. This feature exposes a clone tag when a tag travels in an unexpected direction across readers or misses readers which it should have been read at in an ordered set of readers.

To summarise, the example above illustrated how the presence of clone tags could be exposed in the features constructed from the underlying tag and reader associations. In order for these implausibilities to be apparent in the data, however, the associations between tags and readers clearly needed to exist at various system layers. Having formalised that there was a link between associations and the way attacks were exposed, it may now be possible to identify which RFID systems have these structures, and hence, are conducive to these checks.

8.4 DISCUSSION

This section discusses the implications for solution analysis, in the context of the above example, and makes clear the benefits of this approach by comparing it to those taken by previous work.

The scenario was encoded into the simulation using its application programming interface (API) and executed to produce animation and output data, which provided a repeatable means of analysis. The API enabled the disassembly of a system into its constituent parts to be represented as agents. This imparted the advantage of having one think more clearly about the narrative of attacks in systems, and hence, where particular architectures may reveal attacks. Executing the script led to an animation on screen which depicted the interaction of components at various stages. While it was a simplified view of the system, it did communicate points in the timeline when and where certain associations may form e.g. M:1 association. These were cross-checked with the output data which was a record of component interaction. Whereas previous work on intrusion detection (Koh et al. 2003; Lehtonen et al. 2007a; Mirowski and Hartnett 2007) had to examine their solutions in actual system contexts, the benefit here was that analysis proceeded through multiple-facets of the reference model, e.g. layers and a controlled vocabulary, which were instantiated in the simulator.

The simulator enabled the examination of an attack scenario and insertion of associations from which features were instantiated. The associations represented concepts proposed in Chapter 6, and this chapter took steps towards illustrating their relation to information characterisation. It was seen in the above analysis that attacks could be identified quite easily through these simple features and without the need for additional systems context. From the analysis has come knowledge that systems need to be designed to be conducive to attack detection. Approaching analysis by considering multiple system layers, albeit through simulation, facilitates the identification of when and where in the system it is feasible to identify attacks.

Thus, the ‘whole of system’ approach, made possible by the reference model, lends itself to suggesting ways in which analysis can move from a conceptual model to a system which is simulated. The next chapter will build on this process of analysis by

taking some of the results suggested by simulation and apply these using actual RFID hardware.

8.5 SUMMARY

The focus of this chapter was a 'whole of system' approach to analysis of solutions in the solution partition. To this end, a software program for the simulation of RFID systems was introduced (see Appendix A) to encapsulate this approach. It was then used to explore a simple solution to exposing clone tags in RFID data – illustrating the use of various associations in RFID systems between tags and readers for clones to be exposed. This approach to analysis when compared to previous examples, offers an alternative to analysing a solution in an actual system. Whole of system analysis to solutions means examining solutions across system layers in addition to considering the relationships between other partitions.

The above scenario, examined a location in a system suggested by the reference model as feasible: when tags and readers have interacted to allow for the instantiation of values into constructed features. This location was chosen as it would appear to be the first appropriate place where such attacks could be examined in data. The finding was that simple implausibilities in constructed features may expose attacks such as cloning and pseudo-cloning but that this is reliant on the occurrence of various associations.

The next chapter will use some of the results suggested from this analysis of solutions, to explore practical solutions to the exposure of clones in a M:1 association. The results of that chapter will further reinforce the findings made in this chapter.

Chapter 9

Experiments Facilitated by the Reference Model

9.1 INTRODUCTION

This chapter explores how a 'whole of system' approach to analysis facilitates the exploration of security in actual systems. The previous chapter illustrated how, having built up a framework around security, it is possible to do solution analysis in a system context in simulation. This illustrated how the standard operations, identified in the domain model, influence attack exposure in a theoretical cloning scenario. The results suggested from the previous chapter are explored through laboratory experiments to illustrate the model's role in achieving practical security outcomes.

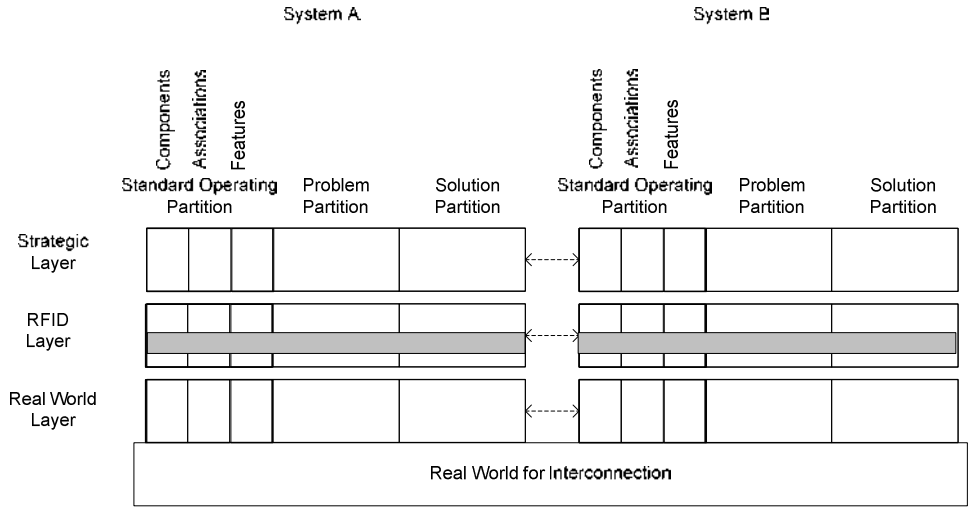

Figure 41 - Experimentation facilitated by the reference model
Simulation results which suggested clones could be identified in a M:1 association have given impetus for exploring the detection of attacks at a reader. The area explored is essentially a segment of the reference model.

Rather than explore all of the simulated results, the results which suggested that was it possible that a segment (illustrated in Figure 41) of the reference model, the M:1 association, may be a suitable location for attack exposure are explored. Simulation suggested that a many-to-one association (M:1) – when many tags are in front of a reader, and using a anti-collision protocol – may facilitate the detection of clone tags. Examining each tag's *time_since_last_seen_at_this_reader* feature in RFID data, implausibilities arose when values were zero.

In exploring this scenario through laboratory experiments, this chapter illustrates that results suggested by the reference model can be confirmed as practical in actual systems and as an additional security solution to RFID.

9.1.1 ACTUAL SYSTEM CONTEXT

In order to explore the simulated results in an actual system context, Electronic Product Code (EPC) Class-One Generation-Two hardware will be used to define a system context.

The significance of this hardware usage was first discussed in Chapter One, and will be refreshed briefly in this section. EPC Class-One Generation-Two UHF RFID standard (EPCglobal 2005), is of importance as, in 2004, it was ratified by the major RFID standards body known as EPCglobal (Roberti 2004). Generation-Two is seen as a significant RFID milestone, as a number of existing standards converged into a single design, enabling manufacturers to produce a globally interoperable technology (ISO 18000-6c), which is seen as the standard expected to become most used in industry. Since then, the surge in item-level tagging, predominantly driven by developments in the apparel industry, has fuelled unprecedented growth in sales of RFID passive UHF EPC Generation-Two integrated circuits (Swedberg 2010b). One of the big retailers, Wal-Mart, has largely been responsible for this increase in sales, and this is a factor seen as continuing to drive growth in EPC Generation-Two UHF tags into the future. This standard may therefore contribute to the eventual widespread adoption of RFID in other industries.

Consequently, a solution which may expose clones in systems that rely on this hardware, would offer an additional security solution.

Work has already been undertaken on examining security in EPC Class-One Generation-Two equipment for the purpose of preventing tag cloning and that work is used as a basis for comparison with the work in this chapter. Juels (2005) defined the problem of tag cloning in technology as the duplication of a tag's EPC stored in tag memory, which is recoded onto another tag that is field-programmable. In the EPC Class-One Generation-Two specification (EPCglobal 2005), an EPC is the number which identifies the entity, to which the tag is or will be attached. It is stored in EPC memory on a tag. It is this data which is obtained off a tag and is stored in a database as a record of which entity was identified at a reader. Consequently, it is this data which is stored in a database after the tag has been identified, attesting to the enity's last known location in the system.

Unfortunately, limited security exists on this type of tag to prevent cloning.

> "…EPCglobal standards prescribe no mechanism for EPC readers to authenticate the validity of the tags they scan. An EPC tag emits its EPC promiscuously, i.e., to any querying reader. Readers accept the validity of the EPC's they scan at face value." (Juels 2005)

Consequently, Juels (2005) proposed that EPC Class-One Generation-Two RFID tags could be reprogrammed to be able to be authenticated by a reader. The idea was that EPC tags, which possess some features toward privacy protection and access control (e.g. the Kill command and password), could utilise these existing functions to construct authentication protocols for tag-to-reader authentication. This would ensure that not only the EPC but also the correct PIN would need to be supplied by an EPC Class-One Generation-Two tag to be authenticated by a reader. This would be a deterrence to cloning attacks in these systems.

Whereas Juels (2005) reprogrammed the tag, this chapter will introduce the concept of reprogramming the reader as a possible way of exposing clones.

9.2 BACKGROUND TO CLASS-ONE GENERATION-TWO ANTI-COLLISION PROTOCOL

The operational characteristics of the EPC Class-One Generation-Two standard is reviewed in this section, and following this, a way of exposing clones at the reader, using the reader's anti-collision protocol is introduced. It focuses on the anti-collision scheme used by the reader to identify multiple tags in the field, as this is the basis for the many-to-one (M:1) association which was examined in simulation in the previous chapter.

Anti-collision is used by a reader to differentiate between signals received from multiple tags simultaneously (Glover and Bhatt 2006). Tags which use an anti-collision scheme facilitated by the reader 'know how to wait their turn when responding to a reader'. Whereas *Singulation* is about identifying individual tags, anti-collision is about regulating the timing of tag responses. (Glover and Bhatt 2006).

The EPC Class-One Generation-Two standard specifies a protocol called the Generation-Two protocol which facilitates anti-collision. During the *inventorying*

stage of this protocol, the reader uses the Slotted Random Anti-Collision (SRAC) method (EPCglobal 2005) and this aspect of the Generation-Two protocol is now reviewed to establish how a reader may expose clone tags.

9.2.1 SLOTTED RANDOM ANTI-COLLISION (SRAC)

This section explains the SRAC method which is used by the Generation-Two protocol to identify tags in the field. This is a relatively brief overview and more information can be found in EPCglobal (2005).

Inventorying is the process of identifying all physical tags in a reader's field (EPCglobal 2005). A single inventory cycle results in the obtaining of each tag's EPC, however, as will be discussed, the EPC is tangential to the actual identification of tags, and under this relationship, clone tags may be identified, through implausibilities which arise in the inventory process. It is worth noting that tags can support inventorying by up to four readers simultaneously through the use of *sessions*. However, the process undertaken by each reader, (and the states maintained by each tag in a session), is essentially the same. Thus, this section just explains how tags are inventoried in a single session.

During a session, tags maintain an inventoried *flag* for an inventory cycle, which can have a value of A or B. At the beginning of every inventory cycle, a reader chooses to inventory tags which are in either the A or B state. After obtaining the EPC of a tag, the reader issues a command that causes the tag to invert its inventoried flag for that cycle – preventing it from responding until the next inventory cycle. The process of identifying a tag population is complete once all tags in the field have been transitioned to the inverse state they began in. For example, all tags which started in state A are moved into the B state, and the next round, by the same reader, will begin in the B state until all tags are moved into the A state. For a tag, this process is repeated until it moves out of range or the reader is instructed to stop inventorying the field. Thus, once a tag has been identified in an inventory cycle, it should not respond until the next inventory cycle, provided the reader has been setup for this purpose.

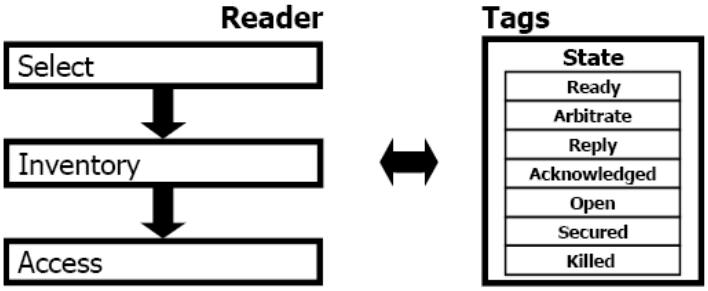

Figure 42 - Reader and tag interactions, and tag states
The process of reader and tag interaction indicates a fixed ordering of steps which result in a tag's EPC data being released during an inventory cycle. (EPCglobal 2005)

During the inventory process, when a tag is moving from the A to B state, or vice versa, a reader proceeds through a series of *states* to obtain a tag's EPC and these states are now reviewed. The process is highly complex and many of the tangential issues leading up to the release of EPC data, such as how tags contend for slots in the protocol, are given brief treatment.

When a tag is first energised by a reader's signal, it enters the *ready* state. This state is a holding state for tags that are neither killed nor currently participating in an inventory round. When a tag, in the ready state receives a *Query* command from a reader, it will draw a q-bit number from its random number generate (RNG), load this number into its slot counter, and transition to the *arbitrate* state if the number is nonzero, or to the *reply* state if the number is zero. In the reply state, it can respond with its EPC. Conversely, in the arbitrate state, a tag decrements its slot counter every time it receives a *QueryRep* command, transitioning to the reply state when this counter reaches zero. Figure 42 illustrates the sequence of states a tag and reader can move through.

Once in the *reply* state, illustrated in Figure 43, a tag will backscatter a 16-bit random number (RN16) to the reader. If the tag receives a valid acknowledgement (ACK) from the reader, the tag will transition to the *acknowledge* state. If the tag does not receive an ACK, it will transition back to the *arbitrate* state, whereby the process of counting down starts over.

Having received an ACK, in the acknowledgement state, the tag will backscatter its EPC data. Also from this state, the tag can move into any other state except killed. If for example, the tag has a password set on the Kill command, the tag can transition

into the *open* state which will subsequently allow additional commands to be issued to the command.

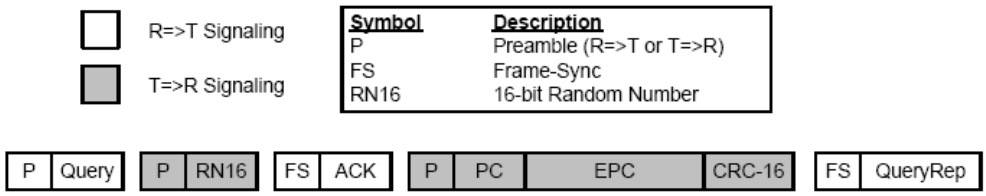

Figure 43 – The 'reply' state for a tag
In the reply state, a tag will firstly establish a relationship using an RN16 value before a tag releases its EPC. (EPCglobal 2005)

Critically, however, this is the point at which the tag has released its EPC to a reader. It is also at this point, where the reader may instruct the tag to invert its inventoried flag, for example from A to B, or B to A, to prevent further communication for this inventory cycle. In Figure 43, the T=>R Signalling states are important as they illustrate that a tag will firstly establish a connection to a reader based on its RN16 value, generated by its RNG on its integrated circuit, then the tag will release its EPC value. Thus, it seems likely that each response in an inventory cycle pertains to a physical tag, and the EPC value is tangential to the data recorded about this association.

9.3 EXPOSING CLONE TAGS

When considering the above process of inventorying tags, it seems likely that; if every tag responds only once per inventory cycle, and a tag response is linked to a tag's RN16 value generated by its RNG, then a reader should know exactly how many physical tags are in the field.

Therefore, it seems that identification of a clone is possible by examining all of the EPC's obtained from all tags in a single inventory cycle. If each response obtained correlates with a physical tag, then determining whether an EPC is a clone, is a matter of simply determining if the same EPC value is recorded in a single inventory multiple times.

The annotated version of this assumption, which will be explored in the laboratory, is illustrated in Figure 44.

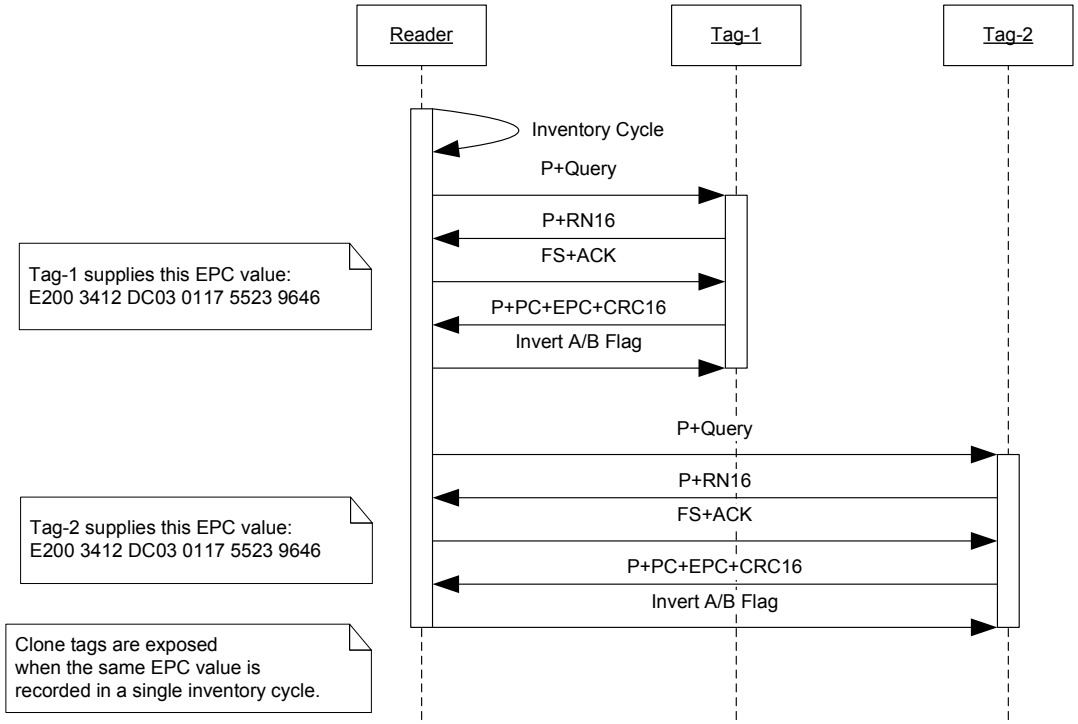

Figure 44 – A single inventory cycle where tags are supply clone EPC values
The clones are exposed when the same EPC value is recorded more than once per inventory cycle, as seen in the above depiction of inventorying multiple tags, adapted from EPCglobal (2005).

The next section sets about determining whether this assumption is correct by undertaking experimentation using EPC Class-One Generation-Two equipment.

9.4 EXPERIMENTAL SETUP AND METHOD

The experimental setup was designed to see if a reader, programmed to undertake and record data from just a single inventory cycle would record data that would allow the existence of clone tags to be easily identified.

To control the reader, commands were issued to it via its application programming interface (API) using a computer attached to the reader. If the reader had allowed for direct manipulation, it would have been possible to reprogram the reader to undertake clone exposure directly at the source of data. For example, by examining responses between the RN16 value and EPC data on the reader. Consequently, readers could be configured during manufacture to use the following approach to expose clones.

9.4.1 EXPERIMENTAL SETUP

The equipment was configured inside a laboratory environment. In an actual system, these experiments would be performed with less of a guarantee of obtaining a response from every physical tag. Thus, this setup may have obtained more effective responses from tags and readers than would be achievable in an actual application environment, in which case, results presented should be treated as optimistic.

9.4.1.1 EQUIPMENT

RFID equipment manufactured to the specification of the Electronic Product Code (EPC) Class-One Generation-Two standard was used for experimentation. The manufacturer of this equipment is Alien Technology; the reader was an ALR-9650 and the tags were Higg-3 tags (AlienTechnology 2007, 2008a, b). This particular equipment operates at 902.75 to 927.25 MHz. As this equipment is Class-One Generation-Two, the reader and tags communicate using the Slotted Random Anti-Collision (SRAC) protocol. The tags are programmable, with a 96-bit memory bank for the tag serial number, which was used when tags were reprogrammed to contain clone EPC's.

9.4.1.2 LABORATORY ENVIRONMENT

The laboratory environment, illustrated in Figure 45, was where the experiments were performed. As the equipment is susceptible to radio frequency interference from surrounding devices, its usage was restricted to a Faraday enclosure, built to a similar specification as that reported by Hashemi (2009), increasing the level of control over radio frequency signals in the laboratory. The enclosure was made from a disused metal filing cabinet, and lined with several layers of aluminium foil. This was intended to provide some basic protection from external radio frequency (RF) interference.

The main goal in using a Faraday enclosure for RFID was to isolate tags from external interference. However, Hashemi (2009) reports that this enclosure design is susceptible to internal interference, whereby internal signals will interfere with one another. However, this problem was assumed to be less of a concern than external interference, outside of the Faraday enclosure.

Figure 45 –Laboratory and Faraday enclosure
The enclosure was a metal filing cabinet lined with aluminium foil, to isolate the equipment from external RF interference and maintain a controlled test environment.

9.4.1.3 EQUIPMENT USAGE

Figure 46 illustrates how tags were positioned inside the Faraday enclosure. The most appropriate alignment of tags was found, through trial-and-error, to be a horizontal alignment with a tag's antenna facing forwards in the enclosure. The most appropriate spacing of the tags was found to be two columns, of ten tags, on a single cardboard sheet. As the number of tags in the enclosure changed, more or fewer sheets of cardboard, which the tags were mounted on, were used. Additional arrays of tags, on cardboard sheets, were found to operate more effectively when they were 3cm apart, at a height of 3cm off the cabinet's surface, and aligned at alternating off-centre positions.

The Faraday enclosure's ability to isolate the equipment from external interference, whilst the equipment was in the enclosure, was established during a calibration phase. This involved two simple tests. Firstly, the reader was put into its *on* state and the enclosure was shut. Some tags were placed on the outside of the cabinet in close to the reader. Secondly, the inverse of this scenario was tested: the reader was mounted on the outside of the cabinet while tags were located in the draw. The outcome was that the reader was unable to read the tags. As no tags were read by the reader in either case, it was assumed that the tags were immune to signals which could disrupt production of data at the reader. However, as no specialist RF signal monitoring equipment was used to gauge potential interferences or dead spots within

the Faraday enclosure, it was not possible to say that interferences within the enclosure did not occur. However, assuming interferences inside the enclosure did occur, as the configuration was consistent for every experiment, the same potential for error would have been consistent across all results.

Figure 46 - The Faraday enclosure was lined with aluminium foil
The foil reinforced the prevention of RF interference emanating outside of the enclosure.

Over such a short distance it was assumed that no apparent difference would occur due to tag positioning. However, in a real system, the prevalence of physical entities to obscure the propagation of tag signals, for example, could interfere with tags. It would be equally applicable to both originals and clones, and thus, not a parameter in allowing clones to be detected. The only way to establish this with certainty would be to examine far more configurations of hardware.

While the above configuration was implemented inexpensively, it provided a reasonable amount of control over external interferences to the RFID equipment, as well as control of the experimentation process.

9.4.2 EXPERIMENTAL METHOD

In designing the experimental method to facilitate the derivation of results, illustrated in Figure 47, consideration was given to the following factors: scalability of results, positioning and usage of tags, experimental repetitions, and formatting of result data.

Scalability of the results considered the size of the tag populations which were used. It is generally accepted that the more tags in front of a reader, the longer it will take to identify all tags due to anti-collision scheduling. In addition, as more tags are introduced, space becomes a limiting factor in situating tags. Consequently, experimentation was conducted using only four different sized tag populations: 1, 10, 50, and 100 tags. Within this range of tags, the derived results can be interpreted as reliable. A summary of the experimental method is illustrated in Figure 47.

1. Experimentation begins when the user initiates the software program on the computer.
2. Software undertakes *y* iterations for current experiment e.g. 15 iterations for tag population of 100 tags.
    a. For iteration *n* of *y* iterations e.g. 5 of 15 iterations
        i. Software instructs user to configure tags in the Faraday enclosure to correspond to the random selection of tags and tag positions it has selected.
            1. User configures the tags in the Faraday enclosure then confirms to software that tags are ready to proceed.
        ii. Software instructs reader via its API to clear all previously held states and settings.
            1. Reader is reset and a new configuration is written to the reader.
        iii. Software instructs the reader via its API to issue a single *inventory* command over the air. The software enters a waiting loop, while the reader obtains tag responses.
        iv. Software assumes, after a five second wait time, no more responses will be supplied by the reader for this inventory command. The software terminates, which terminates the connection to the reader.
    b. Software writes the results for iteration *n* to disk for manual processing.
3. Experimentation ends when *y* iterations are complete.

Figure 47 – General overview of the experimental process
The method could easily be adapted for actual systems by simply reprogramming a reader or some other connected component to examine data obtained per inventory cycle.

The software used for this method to select which tags were used, and to control the reader can be found in Appendix B. It utilises the application programming interface

(API) of the Alien Technology hardware which was used. More information on this hardware can be found in AlienTechnology (2008a) and AlienTechnology (2007).

An experiment consisted of 15 repetitions and in each repetition; software randomly selected a subset of different tags from a group of tags. This process was designed to select which tags were used and where these were to be positioned. As some tags may have been damaged, manufactured differently, or placed in a different part of Faraday enclosure, selection and positioning took steps to ensure results were collected across a variety of different tags. In the results, the selection of tags and positioning is identified at the top of the result set. Of course this process would not actually be undertaken in an actual system where clones were expected to be exposed – this process is only for experimentation purposes.

Recall that commands were issued to the reader via its API on an attached computer. This has the distinct drawback of requiring additional equipment for experimentation; however, if it was possible to directly manipulate reader firmware, then it may be possible to configure a reader during manufacturing to expose clones.

The results for an experimental iteration were recorded by the software. The software did not aggregate the results across all repetitions, instead choosing to present the raw stream of data, so as not to obscure the relationship between a physical tag's response, an inventory cycle, and the reader.

Thus, in an actual system, a reader would assess the validity of tag responses when tags were active in front of the reader but consideration should be given to influences which may be detrimental to results collection.

9.5 RESULTS

The results which suggest that tag responses obtained in an inventory cycle are associated with physical tags and independent of EPC values are now presented. On the basis of these results, a way of exposing clones in an inventory cycle is presented. The full set of results can be found in Appendix C. In each set of results, the following details are listed: repetition number, which physical tags were in the field and where these were positioned, the issuance of the inventory command (get TagList), the results of the first tag stream (which is always the reader's network

information), and finally, the responses obtained from tags. Some limitations on the collection of these results are highlighted and left to be examined in further work.

9.5.1 RESULTS SUGGESTING RESPONSES IN AN INVENTORY CYCLE CORRESPOND TO PHYSICAL TAGS

This section presents results which suggest an association between physical tags and responses recorded by the reader in a single inventory cycle exist. Establishing this relationship is needed in order to realise the implausibility produced when multiple physical tags respond with the same EPC value in a single inventory cycle.

```
Rep:2 of 15
[94]
Have you configured the tags? y/n
Issuing Command:get TagList
TAG STREAM 0
#Alien RFID Reader Tag Stream
#ReaderName: alien2
#Hostname: alien-00074C
#IPAddress: 10.1.1.10
#CommandPort: 23
#MACAddress: 00:1B:5F:00:07:4C
#Time: 2010/10/25 09:26:18.110
TAG STREAM 1
Tag:E200 3412 DC03 0117 5523 9281, Disc:2010/10/25 09:26:18.095,
Finished Rep:2 of 15
```

Figure 48 – Response from one physical tag in the field with a unique EPC
A single response is obtained following the issuance of a single inventory command.

Figure 48 illustrates the result of issuing a single inventory command when one physical tag was in the field. This tag, labelled number [94] in the laboratory, and configured with the EPC E200 3412 DC03 0117 5523 9281, provided a response. The tag response was received at 09:26:18.095 in tag stream one. The fact that a single response was obtained at the reader, is consistent with one physical tag known to be in the field.

```
Rep:4 of 15
[59][88]
[30][19]
[52][76]
[24][61]
[79][94]
Have you configured the tags? y/n
Issuing Command:get TagList
TAG STREAM 0
#Alien RFID Reader Tag Stream
#ReaderName: alien2
#Hostname: alien-00074C
#IPAddress: 10.1.1.10
#CommandPort: 23
#MACAddress: 00:1B:5F:00:07:4C
#Time: 2010/10/25 05:05:15.124
TAG STREAM 1
Tag:E200 3412 DC03 0117 5523 9214, Disc:2010/10/25 05:05:15.089,
TAG STREAM 2
Tag:E200 3412 DC03 0117 5523 9281, Disc:2010/10/25 05:05:15.219,
Tag:E200 3412 DC03 0117 5523 9119, Disc:2010/10/25 05:05:15.219,
Tag:E200 3412 DC03 0117 5523 9295, Disc:2010/10/25 05:05:15.229,
Tag:E200 3412 DC03 0117 5523 9269, Disc:2010/10/25 05:05:15.229,
Tag:E200 3412 DC03 0117 5523 9175, Disc:2010/10/25 05:05:15.239,
Tag:E200 3412 DC03 0117 5523 9187, Disc:2010/10/25 05:05:15.239,
Tag:E200 3412 DC03 0117 5523 9254, Disc:2010/10/25 05:05:15.249,
Tag:E200 3412 DC03 0117 5523 9465, Disc:2010/10/25 05:05:15.249,
Tag:E200 3412 DC03 0117 5523 9258, Disc:2010/10/25 05:05:15.259,
Finished Rep:4 of 15
```

Figure 49 – Response from ten physical tags in the field with unique EPC's
Ten responses are obtained which suggest each response came from a physical tag in the field.

Figure 49 illustrates the result of issuing a single inventory command when ten tags were in the field. The arrangement of the tags in the filing cabinet, as multiple tags were in use, matches the illustration, thus, [59] [88] were positioned at the top of the array in the Faraday enclosure and so forth. This maintains a record of where in the Faraday enclosure each physical tag was positioned. It can be seen that following the issuance of the inventory command, ten unique EPC's were obtained at the reader. These responses span two tag streams, most likely, as the software received the data from the reader in different threads. The fact that ten responses were received is consistent with ten tags being known to be in the field.

In each of these examples a single inventory command was issued and the results collected. For the same tag population, multiple issuances of the inventory command were not applied. That is, this process - an inventory command followed by results collection, then followed by another inventory command and results collection – was

not performed. This would have been a more robust process but was not undertaken as, most likely; it would have illustrated the same associations.

In some examples in Appendix C, it is apparent that the tag populations of sizes 50 and 100 tags have missing physical tags in the results. In some iterations, fewer tags then in the population responded to the issuance of a single inventory command. For example, 90 tag responses rather than the full 100 tag responses. This may be due to interference in the Faraday enclosure, for example. In these cases, it would have been useful to issue multiple inventory commands on the same population for a nominal time period, e.g. 3 seconds, to see whether the reader could detect any additional tags which were not recorded in the first instance. Alternatively, additional RFID readers or antennas could have been used in the same field to increase coverage of the tags, thereby, increasing the likelihood that all tags would be read in the first command issuance. The fact that some iterations produced less than the actual number of physical tags, however, is not perceived to be an issue in this chapter. The data obtained still establishes the association between those physical tags which did respond to a reader's acknowledgement (ACK) command, and the obtaining of EPC values.

Thus, when considering the above examples, it seems likely that the responses obtained during a single inventory cycle are associated with the physical tags which are in the field. These responses will be confirmed as independent of the EPC's in the next section, and therefore, these responses can be to expose clone tags in a M:1 association.

9.5.2 RESULTS SUPPORTING THE EXPOSURE OF CLONE TAGS IN AN INVENTORY CYCLE

The results which expose implausibilities in tag responses, and therefore suggest the occurrence of physical tags with clone EPC values is reported in this section. In the following examples, the experiments were conducted using populations of: 1, 10, 50, 100 tags (results of which appear in full in Appendix C). In these experiments tags were configured with the same (0102 0304 0506 0708 090A 0B0C) and therefore, tags responded with the same EPC value to the reader during the inventory cycle. On this basis they constitute clone tags under the definition proposed by Juels (2005).

```
Rep:2 of 15
[7][5]
[6][4]
[2][1]
[10][3]
[8][9]
Issuing Command:get TagList
TAG STREAM 0
#Alien RFID Reader Tag Stream
#ReaderName: alien2
#Hostname: alien-00074C
#IPAddress: 10.1.1.10
#CommandPort: 23
#MACAddress: 00:1B:5F:00:07:4C
#Time: 2010/11/20 05:41:11.411
TAG STREAM 1
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.396,
TAG STREAM 2
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.516,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.516,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.526,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.526,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.536,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.536,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.546,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.546,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:41:11.556,
Finished Rep:2 of 15
```

Figure 50 – Responses from ten physical tags in the field
The results suggest that obtainment of the same EPC in a single inventory cycle are indicative of clones in the field.

Figure 50 and Figure 51 both illustrate responses captured by the reader, following the issuance of a single inventory command on a population of ten physical tags which contained the same EPC value. The results support the associations found above: that responses are associated with physical tags as ten responses were collected. The fact that these responses all use the same EPC value does not appear to have influenced tag identification at the reader. When considering these examples, it seems likely that responses obtained by a reader, in a single inventory cycle, are independent of the EPC value – which is as should be expected following the operation of the protocol (EPCglobal 2005).

```
Rep:12 of 15
[3][2]
[5][6]
[10][4]
[8][9]
[7][1]
Issuing Command:get TagList
TAG STREAM 0
#Alien RFID Reader Tag Stream
#ReaderName: alien2
#Hostname: alien-00074C
#IPAddress: 10.1.1.10
#CommandPort: 23
#MACAddress: 00:1B:5F:00:07:4C
#Time: 2010/11/20 05:42:11.481
TAG STREAM 1
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.456,
TAG STREAM 2
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.586,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.596,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.596,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.606,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.606,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.617,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.617,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.626,
Tag:0102 0304 0506 0708 090A 0B0C, Disc:2010/11/20 05:42:11.626,
Finished Rep:12 of 15
```

Figure 51 - Ten more physical tags in the field as clones
The additional repetitions confirm that, indeed, clones can be revealed when duplicate EPC's are observed in a single inventory cycle.

Consequently, when considering the relationship between responses and physical tags; clone tags are trivially evident on the basis that multiple EPC's of the same value were obtained in a single inventory cycle. It is not possible that the same physical tag could have solicited the same EPC value multiple times in a single inventory command.

Thus, to expose the existence of clones in the field, a reader need only check to see if multiple instances of the same EPC are obtained in each inventory cycle.

9.6 DISCUSSION

This chapter has considered the reprogramming of a reader via its application programming interface (API) on a computer for exposing clones. This approach is compliant with EPC Class-One Generation-Two UHF RFID equipment. The process was shown to work for a many-to-one (M:1) association, when many tags are active in front of a reader which uses the Slotted Random Anti-Collision approach. As the

work was suggested by the reference model, and visualised in simulation results, the implications of these findings for a 'whole of system' approach to security are briefly discussed.

Whereas Juels (2005) illustrated that the reprogramming of read-to-tag protocols of authentication protocols can be achieved, the approach proposed in this chapter does not authenticate tags to a reader – rather it exposes clone tags after these have been cloned and introduced back into the system. In the proposed approach, a reader validates the authenticity of a tag on the basis that its EPC data is not already in use by another tag at the reader at the same time. Clone tags which are active in the same field at the same time are therefore revealed to the same end that an authentication protocol would identify the duplicity of EPC's amongst a group of tags. Thus, clones which are travelling in a group, which appear at a reader, could be exposed by this method, providing an additional security solution to that proposed by Juels (2005).

A potential limitation on this approach working is that clones need to appear in the same tag group, and at the same reader in time. Consequently, consideration should be given to what constitutes the 'tag group' and the 'read zone'. The Class-One Generation-Two standard supports tags interacting with up to four readers at a time via sessions, thus, a relatively large perimeter could be defined using multiple readers spread out over a large physical area to define a large M:1 zone. Moreover, other natural areas in which many tagged entities are likely to be in close may be candidate areas for the validation of EPC data. For example, the recursive M:1 structure which may be prevalent on palletised products, tagged to the item-level comes to mind. In Chapter 6, the advantage of having identified and modelled this association, in addition to putting into the vocabulary, is that it is now possible to identify it in systems.

The investigation into exposing clones at a reader was suggested by simulation work conducted in the previous chapter. Having looked at the problem of cloning 'whole of system,' simulation suggested the possibility of examining data directly at the reader. This was a different approach when compared to work on intrusion detection systems by Mirowski and Hartnett (2007) which had previously attempted to look for implausibilities using historical data, and data collected throughout different

locations. When this example is considered, it seems likely the M:1 approach has the advantage of not requiring any additional context for attack exposure.

Looking at the method 'whole of system' suggests that filtering of data may influence how successful this method is. In the results it is apparent many data records are obtained in a few milliseconds per inventory command. The usual tendency would be to filter out all records which pertain to a single EPC to reduce the amount of data entering a database. However, this would mean that clones would not be detected if these individual records were aggregated or filtered out. This suggests the need to perform plausibility checking prior to data filtering.

In essence this chapter has been about how a 'whole of system' approach to analysis, made possible by the reference model, helps in identifying practical security. When layers and partitions are considered, and analysis occurs through these using systematic methods, where it was feasible to *reprogram* a system to expose clone tags was identified. Simulation assisted the search towards this discovery, and experimentation confirmed the results. This solution location was reached when the problem was approached through the reference model.

When considering the above outcomes, facilitated 'whole of system' via the reference model, the advantages which this analysis approach imparts in encouraging the exploration of practical security are evident.

9.7 SUMMARY

This chapter began with an exploration of results suggested by simulation work reported in the previous chapter, and proceeded to confirm these results as a solution to tag cloning exposure using EPC Class One Generation Two hardware. The work completed here serves to confirm the reference model's ability to suggest practical security results while also introducing an important additional solution to clone tag exposure in systems

The simulation results indicated that in a M:1 association, when multiple tags were active in range of a reader, anti-collision may expose the implausibility of tags which contain duplicate EPC values. The simulation model was based on the domain model which was enumerated in Chapter 6, and experimentation was examining

associations of the model which were containing cloned tags. Upon the finding that clones could be identified in a M:1 association, experimentation commenced with EPC Class-One Generation-Two hardware as this is reported as increasingly widespread in industries (Swedberg 2010b). A cloning solution here may assist limit clones in these systems.

The solution which was explored in this chapter has the advantage of being relatively simple, and potentially implementable on existing hardware. It could be an addition to existing solutions which operate within this part of an RFID system, making it an important contribution to RFID security.

When considering the process of arriving at the solution reported above, these results would not have been evident if a 'whole of system' approach had not been applied. Having explored the 'solution partition' through the simulation model in Chapter 8 the search for a means of exposing clones via tag and reader associations was identified. This exploration process improved the search for this solution through animation and output data, along with various concepts contained in the model. The relationships between layers and partitions led to the identification of a potential location for attack exposure: the reprogramming of a reader to expose clones.

This chapter ends with the thought that architecture based solution development, which is what reference models encourage (Mišic and Zhao 2000; Fettke and Loos 2003), discussed in Chapter 4, has been illustrated in this chapter through the contribution of a practicable solution for EPC Class-One Generation-Two systems which was arrived at through the reference model.

Chapter 10

An Application of the Reference Model: A Case Study

10.1 INTRODUCTION

In this chapter, a 'whole of system' approach to security analysis is illustrated by applying the complete reference model, illustrated in Figure 52, to the specific example of a pharmaceutical supply chain, and the results are compared with those which may be derived using previous work which took a more localised approach (Rotter 2008; Mitrokotsa et al. 2010).

Pharmaceutical supply chains are complex systems in which RFID is integrated across a number of locations called *custodians*. RFID is used for the purpose of producing information suitable for electronic pedigrees which are essentially documents each containing a history of a drug's movement through a supply chain, and are used to validate the drug as authentic (or for recall purposes). In examining the security requirements of this specific example, the reference model's ability to be used for security analysis 'whole of system' is revealed.

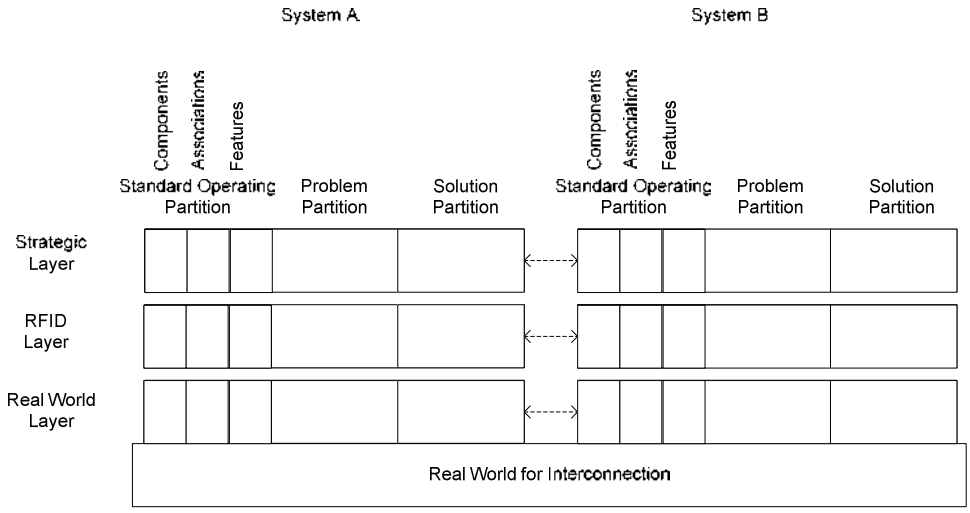

Figure 52 - Validation of the complete reference model via a case study
By applying the reference model to the specific example of a pharmaceutical supply chain, this chapter illustrates the benefits of the 'whole of system' approach via the reference model.

Recall from Chapter 4 that validating a reference model via a case study is a appropriate approach to take (Fettke and Loos 2003). In order to illustrate the rigour with which the analysis of the model will be scrutinised via a case study, one should understand the complexity of pharmaceutical supply chains. A brief background is now provided for this purpose, before outlining the organisation of this chapter.

Since the United States Federal Drug Administration (FDA) recommended the use of Radio Frequency Identification (RFID) in their 2004 report, *Combating Counterfeit Drugs* (FDA 2004), pharmaceutical supply chains are increasingly using RFID as a source of data for *electronic pedigrees*. Electronic pedigrees try to prevent the introduction of counterfeit drugs and diversion of legitimate drugs. These are two ongoing problems for pharmaceutical supply chains (Howe et al. 2007). As the RFID system is integrated with the pharmaceutical supply chain to support the information goals of the electronic pedigree, it is important that the data sourced from the RFID system is valid.

The importance of treating RFID security requirements using a 'whole of system' approach is apparent, when considering the RFID system trialled by the prescription drug manufacturer *Purdue Pharma*. Their system was one of the earliest known RFID-based electronic pedigrees. News sources report that the Purdue Pharma electronic pedigree matched each bottle of the drug OxyContin, a popular painkiller, with a corresponding electronic record detailing the drug's movement through a pharmaceutical supply chain to form an electronic pedigree (Wasserman 2005). News sources report that the Purdue Pharma electronic pedigree sourced its data from an existing RFID system which was monitoring OxyContin that was shipped to the drug wholesaler H.D.Smith (O'Connor 2007). This RFID system allowed H.D.Smith to authenticate the drug by authenticating electronic pedigree information using digital signatures; making sure a bottled drug's pedigree serial number matched the corresponding Electronic Product Code (EPC) on the bottles tag.

When considering the above example, two issues highlight the need for 'whole of system' analysis:

- As an electronic pedigree is a *data file* (Grasso and Cole 2006), changes in different parts of the system, such as those stemming from cloning, can affect RFID data, and hence, the accuracy of the pedigree. Thus, these changes need to be controlled to ensure accurate pedigree data.

- As an electronic pedigree should exist for each individual drug bottle, rather than an entire batch of drugs, RFID will need to maintain its operational advantages when compared to the barcode. For example, the ability to

identify individual tags and tags through packaging - without slowing down the supply chain. In Chapter 2 it was established that the choice and implementation of security has ramifications throughout the RFID system. Thus, where security is located is of importance in achieving a balance between RFID and security.

In accepting that a pharmaceutical supply chain is a complex system, applying a 'whole of system' approach to analysis of security in this system should allow the extent of the model's capabilities to become evident during the deliberation process.

To this end, the chapter presents an analysis of security in the specific example of a pharmaceutical supply chain. Background information which provides a systems context has been first reviewed, which suggests that this system is relatively complex, and therefore, has the potential to utilise all facets of the model. It also provides a system context for analysis in the problem and solution partition. Following this, each partition of the reference model is used to analyse this system. Elements relevant to the analysis are encoded inside a partition, and later, all of the partitions are integrated. During this analysis comparisons are made between the outcomes achieved and those which would be achieved if previous work was applied to the same problem. The deliberative process of analysis illustrated in this chapter is the basis for validating the model's usage for complete systems.

10.2 ANALYSIS OF THE PHARMACEUTICAL SUPPLY CHAIN

In this section, the reference model has been adapted for operational use. To this end, the operational use of the problem partition and solution partition are overlayed on top of the domain partition using the system layers. Overlaying attack sequences over the domain partition depicts the underlying context needed to realise where in the system pharmaceutical systems are vulnerable to attack. Similarly, the solution space is overlayed against the problem space to illustrate its relation to attacks. Essentially, the same model, just used slightly differently when compared to previous chapters.

While the elements which are instantiated into each partition of the reference model are generic, to show how an actual RFID system in a pharmaceutical supply chain

can be discussed through the reference model, the RFID system trialled by Cardinal Health is applied as the specific case study example. News sources report that Cardinal Health trialled the use of RFID starting in June 2006, to track and trace items in its pharmaceutical supply chain (Bacheldor 2006b). Cardinal Health manufactures pharmaceuticals for nine of the world's top ten drug companies and distributes one-third of all pharmaceutical, medical, lab and surgical products in the United States. Item-level tagging for an electronic pedigree was trialled to help improve the safety of medicine and other health-care products as they move through the supply chain in this system. More information about this specific example will emerge in conjunction with generic pharmaceutical supply chain conditions in the following analysis sections.

The reference model is now used to analyse this system with the outcomes of a 'whole of system' analysis presented for each partition.

10.2.1 STANDARD OPERATING PARTITION

In pharmaceutical supply chains, the RFID system can be thought of as a series of interconnected RFID systems. Each subsystem is situated within a custodian's location, to act as a source of part of the information for an electronic pedigree. In this section how the layers and partitions assist in encapsulating relevant information for analysis is illustrated.

To make a distinction between this work and previous work by other authors; previous work has organised the elements of systems into layers (Mitrokotsa et al. 2010) or has appeared to provide a mutually exclusive view of domain structures using several system properties (Rotter 2008). These appear to capture a system's static elements; however, they overlook the influences of system elements on the integration of attacks throughout the system.

Conversely, the reference model organises components across the layers of the reference model and illustrates their interconnections. The pharmaceutical supply chain is modelled at the real world layer, the RFID system at the RFID layer, and the electronic pedigree at the strategic layer. The reference model abstracts, from these layers, components into the partitions; the associations between these components, and the information which can be inferred from their associations. The difference

from previous work is that the layers and partitions will be shown to lead to a more effective view of the system - from the real world layer's most concrete elements, to the strategic layer's most abstract elements - allowing for a system's nuances to be taken into consideration in analysing security in the next two partitions.

Figure 53 depicts two custodian locations using the reference model within a *generic* pharmaceutical supply. These generic custodians are: a chemical company, and a drug manufacturer (Koh et al. 2003). To show that these RFID systems are related, these share the common *real world for interconnection* – the physical world where products are distributed via shipping and receiving methods. At the real world layer for interconnection, product exchanges can involve various types of *shipments* such as the products to be distributed from the chemical company, to the drug manufacturer. Other custodians can be modelled, but for brevity, only two have been depicted. Also, the data flows between elements have not been modelled as these can become very complex. The rest of this section will discuss how the layers and partitions enable a effective view of the system's *standard operations*.

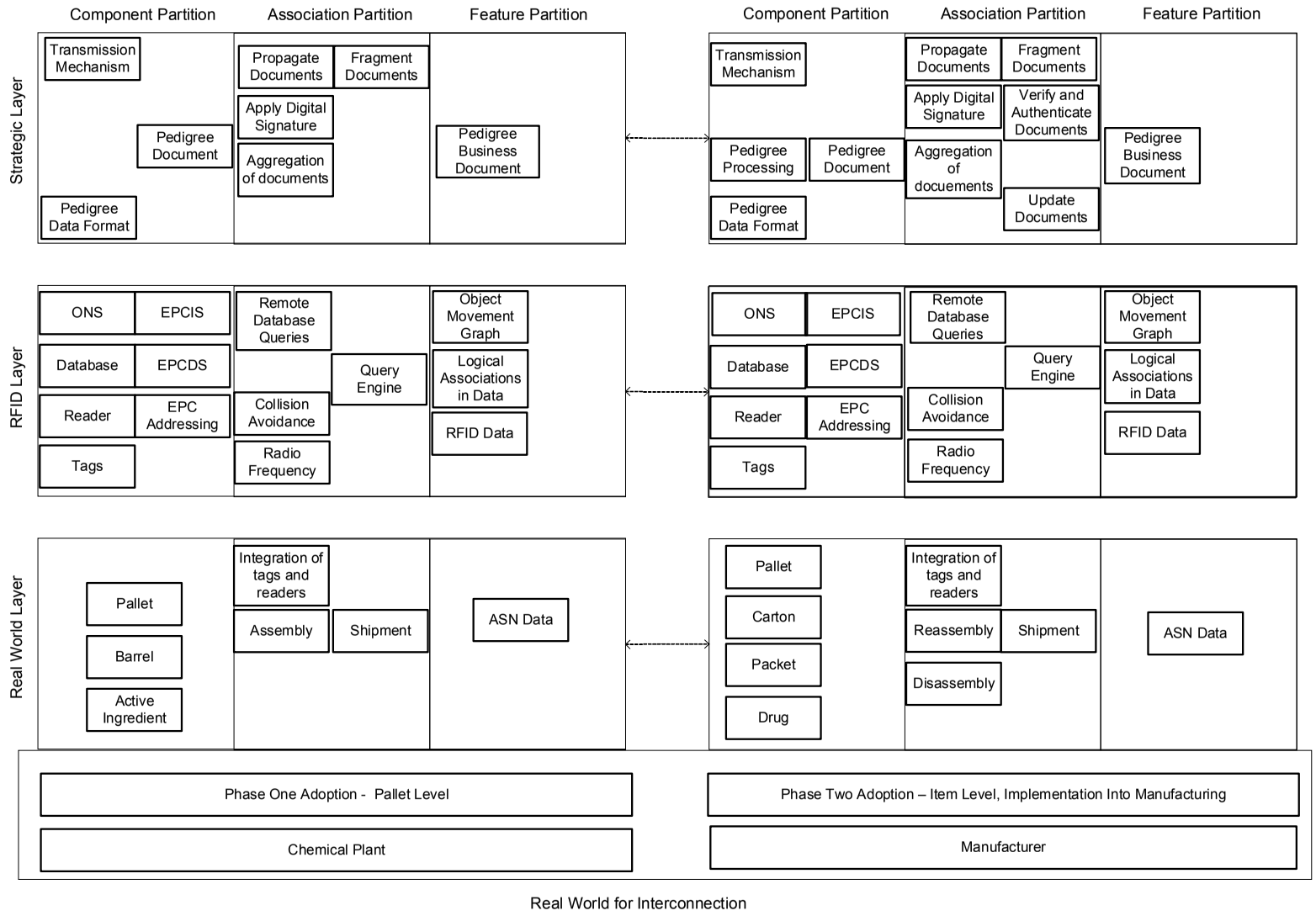

Figure 53 – Standard operating partition of pharmaceutical supply chains
The elements have been organised into partitions and layers, making specific the domain model proposed in Chapter 6.

10.2.1.1 STRATEGIC LAYER

The strategic layer contains the electronic pedigree which establishes a secure chain of custody of *pedigree documents* shared between custodians in the pharmaceutical supply chain. A design for an industry-wide electronic pedigree has been advocated by the Drug Security Network (DSN) and furthered by EPCglobal's *Healthcare and Life Sciences Pedigree Task Force* (Inaba 2008). This design is modelled at the strategic layer as it captures the use of RFID as a source of information for pedigree information goals in the enterprise.

This electronic pedigree uses three elements: *pedigree data format*, *pedigree processing*, and *pedigree information transmission mechanisms* (Inaba 2008). These elements have been modelled in the respective partitions at the strategic layer of the reference model depending on whether these are: standalone components; involve several components by association; or are inferred from data from component associations. These are now briefly explained to justify their inclusion in the standard operating partition.

The *pedigree data format* represents the physical entity's information, such as the drug, in a format which can be distributed amongst custodians (Inaba 2008). It has supply chain wide scope which allows custodians in remote locations, or across countries, to understand and interpret information about entities. Digital signatures are used by the format to ensure the integrity and non-repudiation of data, and also that it complies with legislation and the need for governments to audit such data. As the format has been incorporated into a de-facto industry-wide specification, now essentially a superset of several formats complying with the electronic pedigree laws introduced by the National Association of board of Pharmacy (NABP) and the US states of Florida and California (Inaba 2008), it has been modelled as a component at the strategic layer in the reference model.

Electronic pedigree processing authenticates an *electronic pedigree document*, validating the transactions of previous custodians from the document before the product arrives (Inaba 2008). When physical products arrive, pedigree processing verifies that the products match the electronic pedigree document; and prior to shipping a product, pedigree processing is used to sign the outgoing pedigree, and transmit the pedigree to the next custodian. Pedigree processing is a strategic layer

component which starts at the second custodian's location as this is the first place where pedigree information would be verified.

The *transmission mechanism* of electronic pedigrees is used to transfer data to custodians (Inaba 2008). This can happen in two ways: the *propagating document approach* or the *fragmented data approach*. The *propagating document approach* represents pedigree data into a single document which is appended, re-signed, and forwarded by each successive custodian in the supply chain. As each custodian appends and re-signs the document, a new layer is added to the document, effectively, creating a link between all custodians which can be unwrapped to verify a product's point of origin, thus, it depends on associations forming between custodians. In this case, one-to-many (1:M) associations would be formed as the document is read in sequence between custodians which is why it is in the association partition of the strategic layer.

Conversely, the *fragmented data approach* allows custodians to retain electronic pedigree information for a document in their own database or a third-party database, rather than propagating it down the supply chain (Inaba 2008). Although this produces smaller pedigree documents, it increases the amount of network traffic to source pedigree information. It also means that a custodian could modify data after the product has been shipped. When considering this, as a custodian participates with each other through the database, different associations are formed, for example, a 1:M if documents are read in sequence, or one-to-one (1:1) if a single custodian examines a document in time. Effectively these belong to the strategic layer's association partition.

As products may be *assembled* or *aggregated* in the real world layer, individual pedigree documents are combined into a *Pedigree Business Document* (Inaba 2008). As the Pedigree Business Document serves as a wrapper to consolidate the individual pedigree records, it is an inferred information structure, so it has been modelled in the strategic layer's feature space. These wrappers specify information unique to each pedigree document (identifier, version of format, timestamp); and information unique to the product package (drug name, manufacturer/distributor, object identifier, National Drug Code (NDC), manufacturing date, expiry date, dosage form, strength, container size, lot number, parent package object identifier). This

information attests to a component i.e. a drug. However, as additional information is added to the *Pedigree Business Document* prior to it being transmitted to the next custodian - information about the shipper; and transaction data (sale invoice number, date of purchase, quantity by lot number), belongs in the strategic layer's feature space. This information is derived when several entities in the real world layer are associated with each other – such as through aggregation.

Finally, the custodian that is shipping a product, and hence, a business document as well, has to sign the document. Upon receipt of the goods and documents, the receiving custodian validates the digital signature (authentication) and after matching the received products with the pedigree document (verification), they then sign the pedigree to confirm receipt (confirmation). As the underlying movement of these aggregated products may happen repeatedly along the supply chain - the process of transmission, signing, and retransmission at the aggregated form gets repeated multiple times, which is why these elements are modelled in the strategic layer's association space.

To understand how the strategic layer works in an actual pharmaceutical supply chain that uses RFID systems, the Cardinal Health system is now considered. Having trialled the use of RFID at the item-level, Cardinal Health was readying their Sacramento California distribution centre (DC) for California's upcoming pharmaceutical electronic pedigree requirement (Bacheldor 2007). Having sourced information from the RFID layer, they were preparing to collect RFID data for drugs they receive from manufacturers and distribute to customers. They were also going to purge the serial numbers from any pharmaceuticals that were returned to its Sacramento distribution centre and that data was to be used in an electronic pedigree, to document product return, prior to the drug being returned to the manufacturer. As these procedures would be occurring above the RFID layer, in the enterprise, these would be considered at the strategic layer in the reference model.

10.2.1.2 RFID LAYER

The RFID layer contains the RFID system which is a source of information for electronic pedigrees. There are two sides when considering the RFID system: the lower technology elements (which are physical linkages or interfaces to a custodian's location and products); and the upper data elements (derived from the physical

components which interface to the electronic pedigree). In the case of Electronic Product Code (EPC) based systems, these elements have been standardised, however, other proposals for the design of these elements exist, and therefore, a mixture of both are presented in the reference model. These depict how RFID is a source of information for the electronic pedigree. These elements are now explained in order from the most abstract to least abstract following the principle of abstraction expounded in Chapter 5.

RFID data can be distributed between custodians using various infrastructures. One such infrastructure is the *EPC Infrastructure*, proposed by the Auto-ID Labs, and now advanced by EPCglobal as the *EPC Network*. This infrastructure specifies a global schema for the distribution of data through components and associations (Staake et al. 2005). The main components of the infrastructure are: *object name service (ONS)*, *EPC discovery service (EPCDS)*, and the *EPC information service (EPCIS)*. These elements are now listed:

- The ONS is a multi-layered directory service – containing root and local services - which locate information about tag EPC's in a similar manner to that of the internet's domain name service (DNS). The root ONS is the authoritative directory of manufacturers offering information about their products on the EPC network, whereas a local ONS is a directory for individual products of a specific manufacturer. (Staake et al. 2005).

- Next, the EPCIS is used by trading partners, such as custodians, to store and provide access to product information. (Staake et al. 2005).

- Finally, the EPCDS is a directory of addresses for other EPCIS servers to locate data about an EPC, to be located across several databases in order for track and trace to operate. (Staake et al. 2005).

As these components are networked components, most likely accessed over the internet, custodians would associate to these using *remote database queries* to derive RFID data for their electronic pedigrees, which is why this has been modelled in the association partition of the RFID layer.

On the lower side of the RFID layer, the organisation of RFID components can be reflected in RFID data stored in databases (Agrawal et al. 2006). The associations which arose at the component level, such as when a particular reader has read data from a tag can be inferred from the RFID data, and can be transformed into a *movement graph* of associations.

In a *movement graph*, the physical associations are depicted in the data as *entity types*: object, location, and organisation (Agrawal et al. 2006). A relationship '*belongs to'* defines the association a location in the physical world has with a custodian. To represent RFID events, other relationship types exist, such as: observed, assembled, and disassembled. The *observed* relationship represents an event of when an object was seen at a particular location at a certain time and represents the edges of an object movement graph – a trace of where an object has moved. This relation type, together with the relationship type '*belongs to'*, represents the edges of a traceability network, and hence, the associations between custodians.

Associations within a custodian location and the entities it produces can also be represented at a logical data layer as well. The *assembled* and *disassembled* relationships capture associations which have formed between objects - such as hierarchical associations in packaging, for example, packaging of drugs in pharmaceutical supply chains. This is why these have been modelled in the RFID layer's feature space. (Agrawal et al. 2006).

It is possible to build up a data view of the underlying RFID hardware – this was illustrated in a similar manner in Chapter 6 when associations and features were modelled.

Having defined the traceability network's data and logical views in the data, it is then possible to infer new features from these structures by performing *traceability queries* (Agrawal et al. 2006). A *pedigree query* could be used to reconstruct the complete history of an object, whereas a *recall query*, issued to detect the current location of an object, and a *bill-of-material query* asks for everything that is contained in an object in instances of assembly or disassembly. That is why these queries have been modelled in the feature partition of the RFID layer.

Ultimately these logical associations and features arise from the underlying *traceability network* which is made up of all the RFID *sensing locations* containing RFID readers (Agrawal et al. 2006). Sensing locations within a custodian's location are equipped with RFID readers which produce *events* that represent the state of an object at a certain time. As sensing locations are organised in sequence, and entities move in sequence along these locations, an *object movement graph* can be defined as the derivation of data records in sequence – essentially a history of the entity's movement along a particular part of the RFID system. As each custodian has an RFID system which may be contributing to a centralised data repository, each location is effectively contributing a subset of the overall associations within the object movement graph, and hence, the overall entity trace.

However, as object movement and related data can be valuable business information which custodians may be reluctant to share, a system which maintains sovereignty of each query may be restricted. Agrawal et al. (2006) have proposed a *query engine* to support information sharing across multiple organisations at the traceability network layer. Provided the custodians run the query engine platform, they can run global traceability queries; the query engine rewrites the request to obtain data which is available to the custodian, complying with permissions in place at custodian gateway query engines. This shows that in the RFID layer's association partition, even though a physical association may exist between components, not every feature can be inferred by a custodian.

The above examples conclude the upper side of the RFID system in the reference model's depiction of standard operations. Now the lower RFID side is discussed.

In order for these higher RFID concepts to be supported, the radio frequency of tags and readers will influence the establishing of associations and features at the RFID layer's lower side. The *Pharmaceutical Benchmark* has examined the use of the three RFID frequency types in pharmaceutical supply chains (Howe et al. 2007). High-Frequency (HF) can be used close to liquids such as a vaccine vial, while far field Ultra High Frequency (UHF) cannot. However, UHF has a much longer reach than HF, as beyond 12 inches HF does not function, whereas UHF is effective up to 36 inches. In addition, UHF is more sensitive to the orientation of the chip relative to the antenna and is, consequently, a more appropriate technology for capturing tag

data from cartons arranged on a pallet. Finally, Near Field UHF has been proposed as replacing HF and UHF, but is still under development. Thus, the choice of radio frequency will impact on the granularity of tag reads on associated entities in the real world layer. Further complicating matters is the influence the physical contents of the entities has, as mentioned above; some frequencies are impacted by water and metals. As will be discussed, frequency will influence how deeply within an assembled package the reader will be able to identify tags, and hence, infer the contents of the assembled product in containing physical entities such as drugs. Consequently, this impacts on what is inferred in the association and feature space of the data layers of the RFID layer. In essence, radio frequency emanates from readers, antenna, and tag components to enable associations being formed – which is why it has been modelled in the RFID layer's association space, while components appear in the RFID layer's component partition. This analysis via the reference model illustrates the model's ability to capture these elements.

In the Cardinal Health system a single radio frequency has been used. Alien Technology 915 MHz Class-One Generation-Two tags have been attached to individual packages of all brand-name and generic prescription drugs as opposed to using several different frequencies (Bacheldor 2006a, b). This shows that even though a variety of tag frequencies exist and for different purposes, some companies have found that a single radio frequency can read tags across several physical layers of packaging, and also across the entire pharmaceutical supply chain.

The adoption of RFID within pharmaceutical supply chains determines the horizontal production of data – between custodians – and the vertical production of data – within a custodian's location (Bapat and Restivo 2005). RFID deployment also occurs in phases. In *Phase One*, a custodian may conduct a closed-loop pilot to derive a business case for widespread adoption of RFID. By applying tags to a limited number of product pallets, the mandates of downstream custodians may be achieved. In *Phase Two*, there would be an increasing level of integration of RFID into the custodian's business operations, where the technology may be pushed outwards and into upstream or downstream custodians. It may also include increasing the level of granularity of tagging, to the item-level or into production processes, incorporated into a company Manufacturing Execution System (MES), thereby extending RFID onto the *plant floor*. The effect would be an increase in the

identification granularity to a point at which raw materials can be tagged, and the integration of several raw ingredients to form new products could be recorded. This would provide a record of product formulation. As various interactions could be recorded under such a scheme – raw ingredient interaction through to ingredient interaction with plant equipment or custodian locations - this could all be reflected in the electronic pedigree at the strategic layer.

Thus, modelling the RFID layer between the strategic layer and real world layer, enables the interrelationships (which influence the configuration of this technology and which will shortly be illustrated as influential in security analysis) to be considered.

10.2.1.3 REAL WORLD LAYER

The real world layer contains the pharmaceutical supply chain's physical components and processes. These enable the movement of drugs between all custodians to deliver the drug all the way to the consumer. This layer is interfaced to the RFID layer by the physical tags and readers when they are associated with different physical entities in the real world layer association partition. The degree of integration of RFID technology into the physical world is dependent on the level of integration enabled by the phases of RFID deployment, in addition to the physical components and processes.

A product's form can change over the pharmaceutical supply chain. A drug *product* can begin life as an active *raw* ingredient in a chemical plant. The chemical plant ships the ingredient in barrels to the drug manufacturer. The drug manufacturer processes the active ingredient, perhaps transforming it into a solid drug, such as a tablet or pill, at a manufacturing plant. The manufacturing plant may aggregate many individual drug products using layers of packaging or containers, in which event new associations are formed between individual products to derive aggregated products.

The manner in which RFID is integrated with the physical layers which shape aggregated products impacts on the derived data for the electronic pedigree. However, what can be inferred at the strategic layer's feature partition of an

electronic pedigree's documents ultimately may depend on the configuration of entities that interact in the real world layer's association partition.

An example of this multi-layered association and how it influences what can be inferred by the electronic pedigree can be seen in the Cardinal Health system, when the assembly of packages and products varies. Products can be assembled into: foils, blister packs, bottles, liquids, and solids – and at different points along the pharmaceutical supply chain (Bacheldor 2006a, b). Accordingly the RFID technology is integrated into this physical structure as follows:

- The RFID tags are embedded into printed labels at Cardinal Health's Printed Components facility in Moorestown New Jersey. The RFID labels are then transferred to the company's Philadelphia packaging plant, to be automatically applied to the individual product items and encoded with unique serial numbers. Labels are automatically applied to the individual product items and cases, and manually applied to pallets. (Bacheldor 2006a, b).

- Cardinal Health installed RFID interrogator antennas on packaging lines and RFID portals at dock doors at both the Philadelphia plant and a distribution centre in Findlay, Ohio. RFID interrogators at the DC read the tags as the drugs were received and shuffled through picking and packing processes. (Bacheldor 2006a, b).

- Finally, tagged unit-level drugs mixed in totes with non-tagged items were sent to a pharmacy in the Midwest, where an RFID portal at the store's dock door scanned tags as products moved into the facility. (Bacheldor 2006a, b).

Thus, aggregation of entities can change throughout this pharmaceutical supply chain with entities exhibiting different layers of association with each other at different locations – ultimately this integration starts in the real world layer. (Bacheldor 2006a, b).

Although the physical form of products can change throughout the supply chain, an association still exists for all raw materials into end products in the manufacturing processes through the integration of raw ingredients. The requirement is to ensure

that the manufacturing processes can be reflected in the RFID system, in this case, through the integration of RFID technology with entities (Koh et al. 2003). Conceptually there are at least two *information links* which need to be available in the RFID system: *data aggregation* and *data inheritance* (Inaba 2008). *Data aggregation* is the logical equivalent of *item aggregation* or *assembly*, whereas *data inheritance* is the history of the parent data, whereby it is possible to reconstruct the history of an item including any transformations it has undertaken. Aggregation at the logical layer allows a tag associated with an entity which aggregates other smaller entities, at the real world layer, to infer that small entities are also present.

Consequently, these information links occur at the real world layer. The concept of data aggregation and inheritance allows a single tag, fixed to a pallet to be read, for details about each product on the pallet. Associations can be formed sequentially as entities move between custodians, when ingredients are integrated to form a new product, or through the assembly of entities using packaging. Recall, these associations are easier to establish in RFID systems as RFID does not require line of sight contact for identifiers to be exchanged with the company (Inaba 2008).

In the Cardinal Health system, trying to read tags on individual items, to depict aggregated associations of many products packed in cases on pallets to a single packaging entity (a kind of many-to-one (M:1) association in the real world) was found to be very unreliable (Bacheldor 2006a, b) at some custodian locations. Item-level read rates were very low – between 7.8 percent and 14.3 percent as the radio frequency was not able to reach all the tags in a pallet.

To resolve this problem, a process called *inference* assumed the initial read was accurate, and that nothing had happened to change the status of the pallet or cases. With inference reading, Cardinal Health had an electronic record of all those individual item-tag numbers, which become part of the Advanced Shipping Notice (ASN) for that pallet and its cases. The ASN was held in a database and correlated with the tag numbers for the items on that pallet. Inference was used to avoid the problems that resulted if a read at subsequent points in the supply chain was not as accurate as the first read. The process worked internally but externally it was limited as not every custodian received an item-level ASN. Thus, using the reference model,

an ASN has been modelled in the real world layer feature space as it can be inferred through physical inspection of the assembled entities. (Bacheldor 2006a, b)

This illustrates that what can be inferred in the electronic pedigree ultimately depends on different layers and consideration needs to be given to the interrelationships which occur throughout these layers. This emphasises the need to think 'whole of system' when developing a model of a system's standard operations.

10.2.2 SUMMARY

This section has defined the 'system model' in which RFID for electronic pedigrees are situated. While the layers and partitions are static or *declarative* constructs, the elements within these are not static; at custodian locations, and depending on which custodian is modelled, different elements are instantiated. This shows that the concept of *context*, at least for pharmaceutical supply chains, is not a static concept; rather, it is necessary to realise that context changes throughout a pharmaceutical supply chain.

To reiterate the distinction made from previous work, Mitrokotsa et al. (2010) have localised security to each layer, and Rotter (2008) has localised security to two system properties, this section has highlighted the interrelationships which occur throughout a system, and in doing so, broadened what can be considered in security analysis. As it occurs across horizontal system layers, and between elements of each layer, security needs to consider these effects in order to draw security analysis for actual systems.

In the next sections, this will be shown to be the essential basis for a 'whole of system' approach to security in this RFID system.

10.2.3 PROBLEM PARTITION

RFID systems that operate in pharmaceutical supply chains are vulnerable to a variety of attacks. As RFID is integrated within pharmaceutical supply chains, the pharmaceutical supply chain influences the configuration of the RFID system. Therefore, a 'whole of system' approach will now be illustrated to facilitate consideration of threats in relation to actual system elements.

Previous work has examined threats at a single layer or at a single RFID component (Mitrokotsa et al. 2010), and other work has suggested that attacks can impact other components indirectly (Rotter 2008). Previous work, has however, appeared to overlook how the attacker would need to attack different elements of a system to attain their attack goal, whilst conforming to the system's context.

In this section, the reference model uses the system context to situate *attack sequences* from Chapter 7. It builds on this work which has illustrated how systematising attacks imparts knowledge of which solutions are more effective for a system's context. In adapting this work, some changes have been made to the way attack trees are organised. Whereas the attack trees originally appeared as a hierarchical structure, these now appear across the layers and partitions depending on where certain elements were situated in the context of the domain partition. For example, an attack against a tag is modelled in the RFID layer component partition, whereas an attack against the radio frequency signal between the tag and reader is modelled in the RFID layer association partition. Thus, attacks are now aligned to the system context using the same layers. This improves on previous work which was reviewed in Chapter 3, which did not appear to make explicit the relation of attacks to the synergistic effects or interrelationships in systems in an individual model.

A number of assumptions have been made here, affecting the way an attack could arise in a pharmaceutical supply chain RFID system, and these are listed:

- An attack goal is only attained when the attacker has engaged a number of attacks across different layers and partitions. The domain partition exists across all layers and partitions; consequently, attacking the electronic pedigree would involve attacking the pharmaceutical supply chain as well as the RFID system. Thus, the strategic layer is generally indirectly targeted by an attacker.

- As an RFID system in the pharmaceutical supply chain is contained within each custodian's location, the *attacker* has been contained to within a custodian's location. Attacks between custodian locations can arise but are not modelled for brevity.

- In chapter 7 the attack trees were originally organised by system type – authorisation system and monitoring system – and with each system having different attack goals as their information goals are different. This RFID system distinction is maintained to show that pharmaceutical a supply chain exhibits characteristics of both authorisation and monitoring systems, depending on how data is used.

A detailed explanation of how various attacks work can be sourced from Chapter 7.

10.2.3.1 AUTHORISATION SYSTEM ATTACKER BEHAVIOUR

In Chapter 7, the attack tree depicts attacker behaviour in a generic RFID authorisation system where the attacker's goal is to introduce an unauthorised physical entity into a system controlled by RFID. The adapted attack tree appears in Figure 54.

The attack goal considered here, in a pharmaceutical supply chain, is the introduction of counterfeit drugs. To pass counterfeit drugs off as legitimate drugs, the drugs would need to be assigned an authorised tag - a tag that contains an authorised tag serial number. The drug itself is confined to the pharmaceutical supply chain, and applicable to the underlying manufacturing and distribution processes in place. These would be represented in the *real world for interconnection* as ultimately the movement of drugs is a process which occurs between custodians. Depending on whether the raw ingredients are counterfeited, the actual drug product, or the different packages which contain the drug – at some point along the pharmaceutical supply chain, the *counterfeit* will need to be introduced into the supply chain. The counterfeit then proceeds along the pharmaceutical supply chain and through custodian locations towards the consumer.

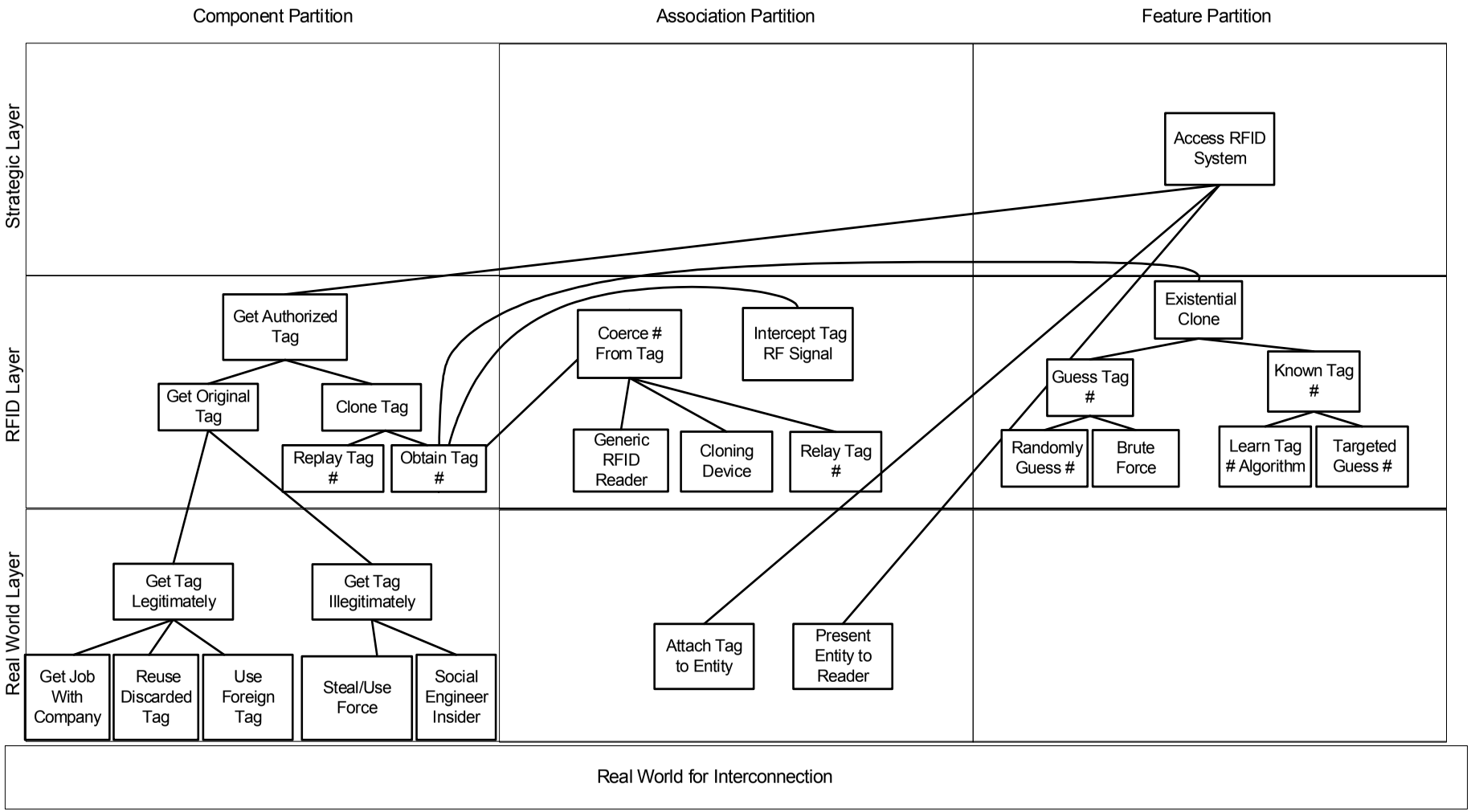

Figure 54 – RFID authorisation system attack tree

Introducing an unauthorised entity is a multifaceted problem as modelled through the layers and partitions of the system. Note, the size of the tree has changed; some nodes have been removed from its original depiction, while the ordering of some have also changed, in order to preserve space.

The attack goal is in the strategic layer feature partition as gaining access to the pharmaceutical supply chain would involve generating false electronic pedigree documents for a counterfeit drug. The data instantiation for these documents, however, occurs as a result of the attacker having obtained an authorised tag, having attached it to a drug entity, and having introduced it at locations along the pharmaceutical supply chain, ensuring that the tagged drug has been presented to readers. Obtaining the strategic layer goal involves attacks across multiple layers and multiple partitions of the system.

The attacker has to obtain an authorised tag for the counterfeit drugs. As a tag is a component it means the attacker must attack the tag in the RFID layer component partition. An authorised tag has to be obtained in order for a legitimate identity to be associated, in the association partition, with the physical drug product - rather than at a later stage spoof a tag's identity to a reader – such as by using a relay attack mounted near the reader (as these attacks are usually performed using specialised cloning devices). A relay attack (Kirschenbaum and Wool 2006) would not be practical as the attacker would not be in possession of the drug. At the point that the drug enters the pharmaceutical supply chain the attacker loses ownership of the drug, and hence tag, which means that the clone tag identity needs to travel with the counterfeit drug.

If the authorised tag is an original tag then the attacker will need physical access to the pharmaceutical supply chain to obtain it. The reason is that original tags would either be attached to drugs already, or in storage and ready to be used on drugs. In either case, physical access to the tags is constrained by the real world layer. Therefore, obtaining access to original tags may mean: getting a job with a custodian; or reusing discarded tags or using foreign tags e.g. cross-contamination (Heydt-Benjamin et al. 2006). Whereas these place the attacker at risk of getting caught, the attacker could proxy out the attack to an insider, who already has access. The attacker could bribe an insider or social engineer an insider; into supplying them with original tags. These all require physical access at the real world layer to components.

Obtaining a physical tag in the Cardinal Health system may not be straightforward (Bacheldor 2006a, b). There are three points where this could be possible – at the

place where they are printed, the *Printed Components* facility in Moorestown New Jersey; or where they are applied to products, the Philadelphia packaging plant; or while the tags are transferred between these locations. However, it is not until the Philadelphia packaging plant that authorised EPC's are encoded onto the tags which mean that the sequencing of authorised tag serial numbers to be associated with drugs is time and location sensitive. These are not enabled until the point at which the tags are about to be applied to the products. Also, in the case of the system deployed out of Sacramento California, there is a clear process in place to purge the serial numbers from any pharmaceuticals that are returned in addition to returning the drug back to the manufacturer, making it difficult to reuse tag serial numbers. Thus, a benefit of a 'whole of system' approach, applied to this scenario, is that such interrelationships can easily be considered during analysis.

Obtaining a clone tag may require encoding an authorised tag serial number onto a reprogrammable tag once the attacker has obtained a valid tag serial number. As the tag serial number, in most cases, is not physically associated with the component, such as being encoded onto the packaging, the attacker can obtain the tag serial number in the association space or feature space at the RFID layer. In the association space the attacker can coerce an authorised tag into revealing its authorised tag serial number using a generic reader. The attacker could also intercept transmission between the authorised tag and a reader if they are in close physical proximity.

In the Cardinal Health system, the possible locations of attacks in the supply chain depends on limitations on tag read range and the way that packaging constrains read performance. A complication arises if tags on the entities are to be cloned as item-level read rates in the scenario vary. It is reported that items in totes – plastic containers filled with drugs - had a read rate of 99 percent to 100 percent, whereas individual items packed into cases on pallets were very unreliable – item-level read rates were between 7.8 percent and 14.3 percent (Bacheldor 2006a, b). Totes were used in the distribution centre (DC) whereas pallets were used prior to the DC. This means that the attacker would need to target tags at the tote level in the DC to obtain tag data, thereby, reducing the entry point of their counterfeits to later in the supply chain, and hence, production of valid pedigree data to validate their counterfeits as legitimate.

Conversely, the attacker can obtain an authorised tag serial number from the RFID layer's feature space, taking advantage of associations which are inferred in the data. While the association space represents the link between components, as the tag serial number is not physically attached to a tag component, or the association which arises between tags and readers, the attacker could *infer* the tag serial number in the feature space at the RFID layer. Existential cloning attacks (Juels 2005) are modelled in the RFID layer feature space as they take advantage of the association between a tag component and the data on the tag. Guessing a tag serial number can be a matter of making a random guess in the number space or brute forcing the number space by iteratively attempting individual tag numbers to determine if they are valid in the system. Another way to obtain a tag serial number in the feature space is by having some knowledge of the tag serial numbers which are authorised in the first place. An insider may be bribed into revealing tag serial numbers or if previous tag serial numbers are known, then existing numbers may be easily guessed.

To complete the attack sequence, the attacker has to take an authorised tag and attach it to a counterfeit drug, and then introduce this tagged drug into the pharmaceutical supply chain. Therefore, attaching the tag to the entity and presenting the entity to the reader are modelled in the association space of the real world layer as they require physical access to the system and they modify the association between entities.

Depending upon the stage at which the attack takes place, in the pharmaceutical supply chain, the depth of counterfeiting, and the extent of RFID in the supply chain, will all influence the degree to which associations in the real world layer need to be modified by the attacker. If every layer of packaging is tagged to the item-level, then the attacker will need to modify each layer, whereas tagged pallets mean the attacker only has to modify the pallet's tag if the entire pallet is counterfeited. If the entity is assembled then it may mean disassembling the entities for the introduction to take place, and then reassembling the package. Once the counterfeit drug has been placed into a packaging configuration – package, carton, and pallet – the underlying pharmaceutical supply chain will move the drug through successive custodian locations onwards to the consumer – in which case, the overall attack goal has been obtained as a valid electronic pedigree would be derived.

For the Cardinal Health system, this would mean the counterfeit drug and tag (whether cloned or original) would need to be inserted amongst legitimate drugs. The insertion would need to occur in sequence to comply with the ordering of the movement of drugs through custodians; at the correct deployment times as tags get *enabled* at the various custodian locations; and also complying with the correct arrangement of other tag serial numbers – if EPC tags are used and assigned in sequence, then nearby tagged entities may be assigned serial numbers within a certain number range.

Thus, the insertion of a counterfeit drug using attacks propagated through the RFID system would appear to be highly involved as RFID is deployed to varying degrees at different layers, and in different custodian locations. This complexity only becomes apparent when consideration is given to attack sequences in actual contexts – the reference model makes this analysis possible. Therefore, when considering previous work (Rotter 2008; Mitrokotsa et al. 2010) it seems less likely one would arrive at this same conclusion as only a few system properties can be considered in these approaches.

10.2.3.2 MONITORING SYSTEM ATTACKER BEHAVIOUR

The chapter on attack trees (see Chapter 7) depicted the attacker behaviour in a generic RFID monitoring system where the attacker's goal was to prevent monitoring systems producing data used to track the location of entities. The adapted attack tree appears in Figure 55.

The attack goal considered here, for pharmaceutical supply chains, is to remove products from the supply chain without the theft being detected. While this at first seems counterintuitive with the overall information goal of an electronic pedigree; denying the monitoring service may be necessary to ensure legitimate drugs and tags do not conflict with counterfeits. As the attacker would need to perform a number of interrelated attacks to remove products, these attacks would occur through the layers and partitions of the domain partition.

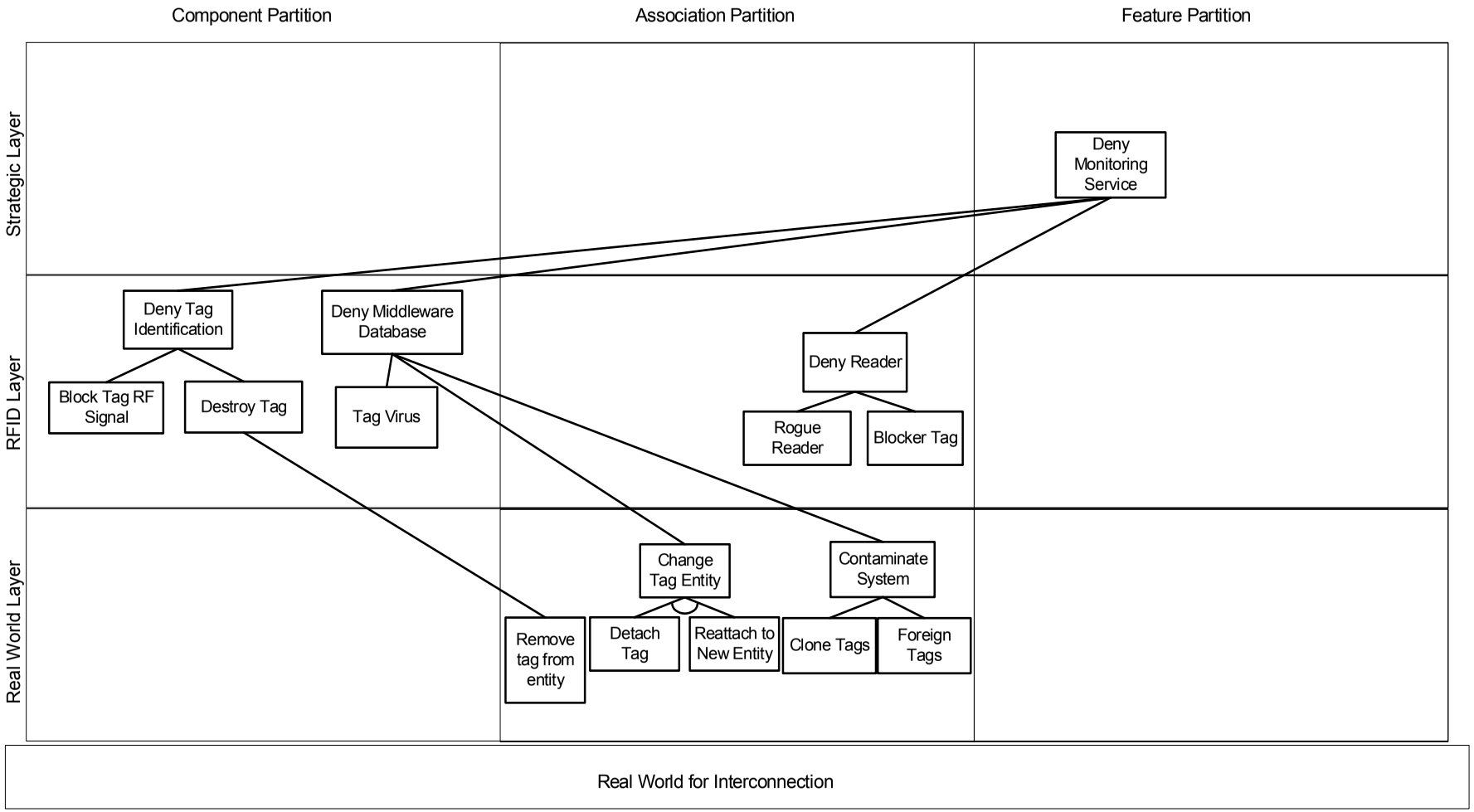

Figure 55 - RFID monitoring system attack tree

Attackers can deny monitoring service in unconventional ways to impact on data stored in the middleware as depicted through the layers and partitions of the system. Note, the size of the tree has changed and some nodes have been removed from its original depiction to fit it into the reference model.

The *real world for interconnection* depicts that the pharmaceutical supply chain is responsible for the manufacturing and distribution of drugs between custodians and onward to the consumer. Thus, the goal to remove products, by first denying the monitoring service from some tags, would occur at the pharmaceutical supply chain layer prior to the introduction of the counterfeit drugs. To this end, the attack goal is in the strategic layer feature partition, as denying the monitoring service of RFID systems, means preventing information being derived between the tag and entity. The severance of the link between the legitimate tag and entity, and any of the successive layers or partitions, would serve to attain this attack goal.

There are a variety of ways the attacker could achieve the attack goal of denying monitoring service. The most obvious way would be to target the tag or reader components, directly in the RFID layer component partition. Denying tag identification would be an attack against the tag component, of which there are several ways of achieving this. The attacker could block a tag's radio frequency (RF) signal using a Blocker Tag (Juels et al. 2003) or by enclosing the tag inside a Faraday enclosure. Either of these would prevent a valid RF signal from reaching the reader as collisions may occur.

However, as the Cardinal Health system uses EPC Generation-Two tags, the chances of a Blocker Tag working may, theoretically, be limited. In Chapter 9, it was explained that Class-One Generation-Two uses the Slotted Random Anti-collision (SRAC) (EPCglobal 2005) whereas the Blocker tag (Juels et al. 2003) is proposed to be used with the Singulation Tree Walking protocol. The former initiates tag and reader transmissions using random numbers, whereas the latter relies on EPC values. Thus, in the Cardinal system, a Blocker tag may be obvious if a reader was reprogrammed to identify the implausibility of the same serial number in multiple inventory cycles (see Chapter 9).

Destroying the tag using the Kill command or by Zapping (Collins 2006) would also be ways of preventing the tag from responding. Removing the tag from the entity would be an attack against the association between the tag and entity at the real world layer as the physical entity, the drug, is involved in the attack. If the entity does not have a tag on it then locating the tag would not locate the entity, therefore, this attack is in the association space of the real world layer.

For the Cardinal Health system, performing denial of service attacks in this manner, may have limited success. The tags at the item-level are difficult to read when they are encased into cases on pallets e.g. item-level read rates were between 7.8 percent and 14.3 percent (Bacheldor 2006a, b). It is not until the drugs are encased into the totes, that they can be instructed through an RFID protocol – which is necessary in triggering the *Kill* password – as read rates are 99 percent to 100 percent at this point.

Moreover, the resultant reduction of missed tag reads due to denial of service attacks from this point forwards appears to be minimised due to the increase in physical system complexity, and hence, added context. The process of *inference* to determine which tags are in range is no longer used, and is replaced with closer-contact tag reads, to the point at which, long-distance reads which were used for efficiency, such as bulk reading of tags in totes, is most likely now one-to-one (1:1) contact to identify tags (Bacheldor 2006a, b). This means that the chances of a tag not being read without the physical presence of the entity or the tag responding, would be noticeable. This demonstrates the benefits of a 'whole of system' approach to security in RFID systems – the reference model enables the consideration of system context in determining which threats are actually relevant.

Attacks on the middleware database, which is a component in the RFID layer, can occur across several partitions and layers. The attacker could use a tag to introduce a virus into the middleware (Rieback et al. 2006). As the tag and middleware database are both components necessary to deliver and execute the attack, these attack sequences are modelled at the RFID layer component partition. A less sophisticated attack would be to target the association between the tag and entity at the real world layer association partition. Detaching a tag from an authorised entity, and reattaching it to a different entity would sever the physical association and also invalidate the association maintained in the database. When the database was queried to locate an entity as the association has changed in the real world layer, the database will be pointing to a different entity.

Contaminating the pharmaceutical supply chain with clone or foreign tags, which share tag serial numbers with authorised tags in the system, would be another way of attacking the middleware, but through the real world layer association space. As the

tag serial number is a pointer to an object in the real world, a reader that had updated the database with an *observation* for an entity based on the reading of a tag associated with the entity, may invalidate the databases data with false reads – as the authorised entity is not actually associated with the clone or foreign tag. In this sense, the layered nature of systems facilitates attacks at the lower layers to invalidate high layer goals; however, as will now be illustrated, this is also where attacks can be revealed.

For the Cardinal Health system such attacks may be constrained by the strict adherence of when and where tag serial numbers are activated. Authorised tag serial numbers are not activated in the system until the Philadelphia packaging plant, at a time when the items are about to be shipped (Bacheldor 2006a, b). As points forward from this location are relying on *inference* to determine what items are on a pallet, if some of these tags were foreign tags, these would not actually produce data until the tote level where pallets are disassembled. This means that foreign tags may not impact on the system until later in the supply chain, at which point, their presence may be noticeable due to: the occurrence of the legitimate tags, non-activation of the serial numbers, or association of tag serial numbers which were not shipped together (specified in the ASN or pedigree data documents).

Finally, the attacker could target the association between the tag and reader in the RFID layer association partition by using a rogue reader or Blocker Tag (Juels et al. 2003). Either of these components, if used, could interfere with the obtaining of tag data. A rogue reader may cause the tag to be preoccupied in responding to its requests thereby preventing it from responding to the authorised reader, whereas the Blocker Tag would respond to the requests emanating from the authorised reader thereby causing it to traverse its binary tree of tag addresses to exhaustion. However, this may not work as EPC Class-One Generation-Two uses anti-collision and not Singulation to identify tags (as discussed above).

In the Cardinal Health system, either way, the ability to associate authorised tags and readers, to establish an association to the entity, may have limited success. There are multiple custodians which are reading tags, and within custodian locations, readers deployed throughout the manufacturing processes, to allow for many potential associations to be formed between tags and readers – thus, 'whole of system'

analysis reveals that these attacks which have been considered to be widespread threats, seem relatively ineffective when considered in the context of this case study.

10.2.3.3 SUMMARY

This section has used the standard operations of systems, in addition to layers, to organise attack sequences against the RFID system. Depending upon the elements contained at different custodian locations, across the layers and partitions, attack sequences vary in terms of the attacks which were sequenced. As domain context can change, at least for pharmaceutical supply chains, so too can attack sequences. This means that different threats will arise at different places as the context varies. Thus, this section has illustrated that not only is a 'whole of system' approach to attacks beneficial in exposing unwarranted risk of some threats, but also realisation of which parts of a system may be more at risk.

This is in contrast to Rotter (2008) which seems to suggest that a system faces a single risk rating, rather than varying degrees of risk rating depending on standard operations and which threats are feasible. Moreover, it is an improvement imparted over Mitrokotsa et al. (2010) as they suggested security attributes like cost or potential damage were localised to a particular layer. The analysis reported above suggests that as system context varies so will the security requirements vary.

10.2.4 SOLUTION PARTITION

In order to understand which solutions may be practicable for this system, 'whole of system' analysis made possible by the reference model is applied to the solution partition. It builds on various analysis outcomes contributed above which have modelled a systems standard operations and attacks. Thus, a robust basis for concluding various solutions is made available.

Conversely, previous work appears to suggest that a direct association between threats and solutions should be followed. Mitrokotsa et al. (2010) suggest that a threat is addressed at the layer at which it occurs, which has the disadvantage of incurring the related attribute values, e.g. cost, at the same layer. Similarly, by localising security analysis to two system properties, Rotter (2008) appears to suggest that systems are limited to being as secure or insecure if they are of a particular system type. That is, *most industry applications* demand low to medium

security. In this section, when all of the above analysis work is taken into consideration, the fact that security requirements vary in different parts of the system will illustrate that actual system analysis is less suited to *localised* analysis – and this is apparent when security is considered through the reference model.

In this section, continuing a 'whole of system' approach, the reference model is used to examine:

- How applying 'whole of system' analysis assists in identifying solutions as feasible.
- How the reference model exposes areas of the system which may be supportive of solutions which do not incur the penalties associated with tag based security.
- How a number of solutions can be implemented to achieve robust defence in this specific system context.

To examine these three areas, RFID intrusion detection techniques, which look for anomalies in RFID data, are examined. Other solutions could have been examined, however, such an in-depth examination of solutions is not needed to illustrate that a 'whole of system' approach is effective at handling such a complex and broad analysis problem.

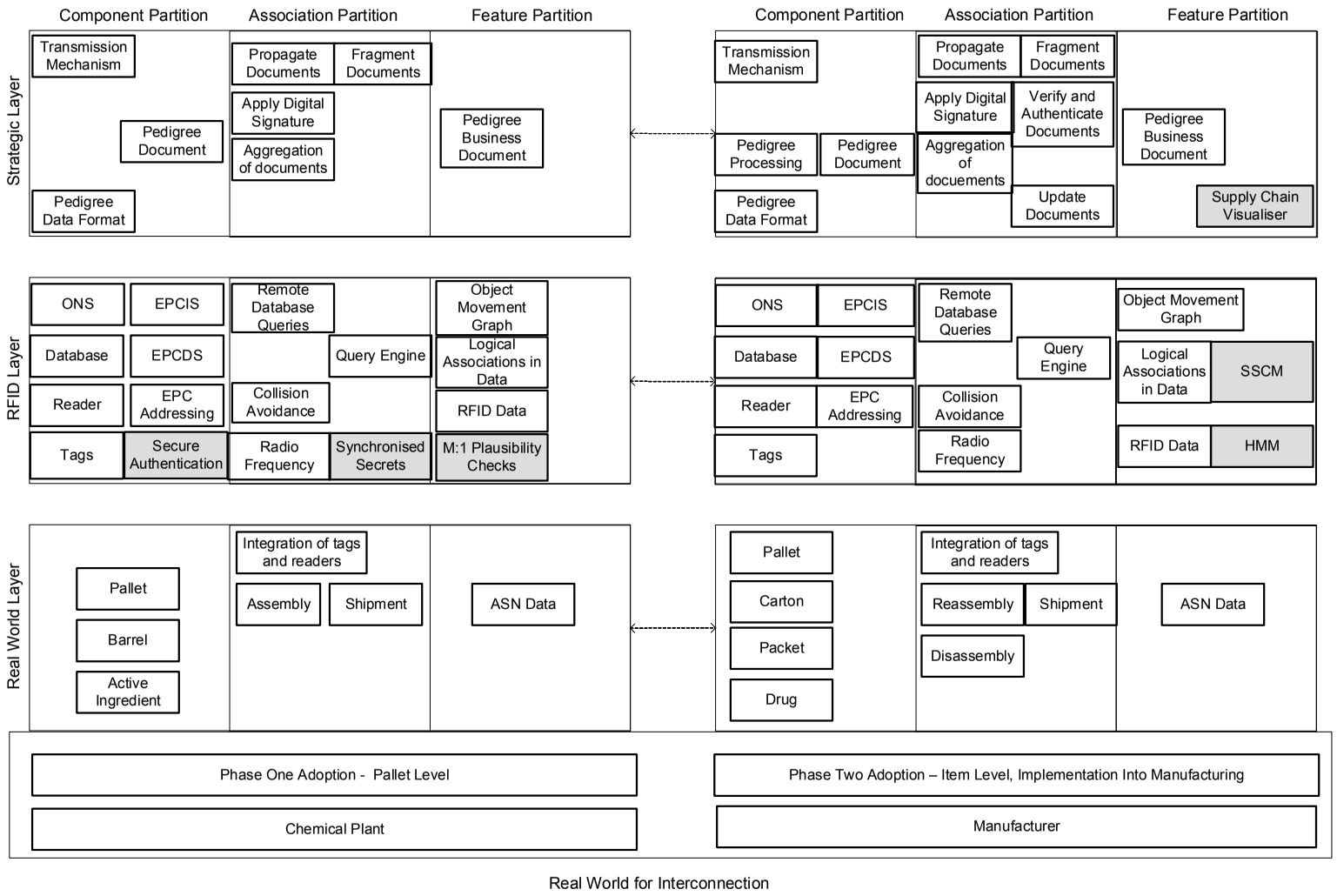

Figure 56 - The solution partition of pharmaceutical supply chains

The solutions have been organised into layers and partitions, in conjunction with the wider context of the domain space to show that some solutions, like intrusion detection, are multifaceted and not confined to a single location.

As counterfeit drugs may use RFID tags – whether clones or originals – to gain access to an RFID system, and hence the pharmaceutical supply chain, to claim the identity of a legitimate drug, a problem to consider is detecting invalid track and trace data from end-to-end of the pharmaceutical supply chain.

A solution could be the *Supply Chain Visualiser* (Illic et al. 2009), a software program which tries to detect counterfeits using plausibility checks, using EPCIS event data and a shipping and receiving model. The plausibility checks are: velocity (tag speed is valid); dwell-time (tag has not remained idle); lifecycle (tag has a logical start and end point); pair-wise shipping and receiving (atomic transactions when tag enters/leaves location); and transition probability (tag routing fits product histories). Detection starts at each root of a product flow and checks each of the currently enabled rules for each time-ordered pair of events, with detected conflicts highlighted as *hot spots* using a colour coding scheme over a map of the supply chain where rule inconsistencies have occurred. As an end-to-end view of the supply chain is taken, this solution would begin in the strategic layer feature space. To show that several custodians would be needed to derive data for the approach, in Figure 56, it has been modelled in the second custodian's framework. This is not to say that a later custodian could not access the solution, it would just require data from several locations before it was useful.

However, to define plausibility checks, elements throughout the domain partition are required. This solution, is therefore not confined to a single layer or partition. The Supply Chain Visualiser relies on EPCIS features which are instantiated through the EPCIS specification. These features are: epcList, eventTime; action; bizStep (shipping, receiving, internal); and bizLocation. As the bizStep feature indicates that an object was received by a custodian, and read at a specific location, a serialised global location number (SGLN) can be resolved into a physical location through geo-mapping coordinates with latitude/longitude values. In addition, plausibility checks rely on other constructed features during a *pre-processing* stage. *Grouped* epcList events are split into *single events* to create chronologically ordered lists for an individual item's flow - identified by its EPC. Time differences, distances, and the movement velocities are then calculated. The derived data from the pre-processing stage forms the basis for the analysis process. Clearly without elements from these layers and partitions, the Supply Chain Visualiser would fail to detect counterfeits.

In the Cardinal Health system, consideration would need to be given to the duration of tags on entities which were monitored by the Supply Chain Visualiser. In the real world layer, entities are assembled at different stages, with tags accordingly applied. At the Philadelphia packaging plant, tags are attached to pallets, cases, packages, and individual product items (Bacheldor 2006a, b). At different stages, entities are removed from these layers of packaging at different points. At the DC entities, for example palletised products, are disassembled and placed into totes. This may mean that the top-level continuum of monitoring the pallet level tag is broken as the pallet is no longer used at this point. The level of granularity in monitoring tagged entities changes throughout the supply chain, affecting the system's ability to have a continuum in the security solution. This may mean that the system's ability to detect counterfeits may vary depending upon the point in the supply chain which is under treatment. This becomes apparent having used the layers and partitions of the reference model to consider the whole system.

As tag cloning may arise at intermediate points along a supply chain, instead of presenting a sequence of anomalies across several custodian locations, a complete track and trace history may not be available. Consequently, Lehtonen, Michahelles et al. (2007a) proposed techniques to use incomplete location information to authenticate products that flow in a supply chain. Assuming that detection of counterfeits is easy if the location of the genuine product is known, conversely, the identification of the counterfeit grows increasingly more difficult as the time since the original observation of the genuine item was made.

Two solutions are possible at intermediate custodian locations: a stochastic supply chain model (SSCM) approach and a hidden Markov model (HMM) approach (Lehtonen et al. 2007a). These solutions both calculate a *transition probability* to signify the likelihood that a genuine product made a transition defined by two events recorded in RFID track and trace data. If the *transition probability* is above a threshold value, the second event is likely to be generated by a genuine tag. As these approaches are taking a partial view of the supply chain, relying mostly on local RFID data sourced by a custodian, these have been organised into the RFID layer's feature partition – but would also require elements from other areas at or below this partition and layer.

For attacks to be detected, training data from a pharmaceutical supply chain was required (Lehtonen et al. 2007a). In their work, it was simulated but in actual systems it would be sourced from the system. Also details are required for the simulation of the supply chain's underlying physical operational characteristics. For example, to simulate incomplete data, the probability that an observation was generated when a product enters a data sharing node was set to 98% and it was assumed that 50% of products that entered the final node were observed, corresponding to capturing and sharing the point-of-sales and point-of-use data. A period of four months needed to be simulated for the distribution of 30,000 genuine products and 900 counterfeits. For an actual deployment of these approaches, this would mean obtaining data from actual custodians to configure the rules. (Lehtonen et al. 2007a)

Consequently, for the Cardinal Health system, calculating a transition probability may be a relatively straightforward matter as it appears as though the pharmaceutical supply chain, and hence RFID system, is relatively sequential from end-to-end (Bacheldor 2006a, b). Although the underlying dynamics of the product flow may vary based on supply and demand, overall, there may be little variation in the way products move in sequence. Thus, anomalies may be quite obvious provided a suitable layer of traceability data was derived from the pharmaceutical supply chain. This realisation comes from 'whole of system' analysis.

Missing RFID data can be caused when interference occurs at the real world layer due to: too short reading times; collision in the air interface; or conductive materials that absorb radio waves. In other causes, the middleware can fail to listen to antennas which are producing data, or *discovery services* at the application layer may not return 100% of the data – perhaps some custodians do not share all of their data. All of these issues will influence how well solutions operate in this context.

In response, Lehtonen et al. (2009) proposed a filtering based technique which corrects incomplete RFID data traces and verifies this filtered data for anomalies. However, to obtain good results, the approach relies on the precise modelling of the physical supply chain. The filtering algorithm is used to *correct* detected missing reads by adding in assumed locations. Reliable detection results were obtained from the simulation of a pharmaceutical supply chain but false alarms were apparent. The

detection rates at the zero false alarm rates were less than 30%, whereas a 95% hit rate at below 0.2% false alarm rate was achieved, however, as the missing reads increase, more training data was required. The work shows that even though missing reads do occur in RFID systems, the use of filtering algorithms can correct erroneous data, meaning these are not as significant a problem as previously thought for detection approaches. Clearly, however, an understanding of the physical pharmaceutical supply chain is needed in order for this approach to work. Thus, reinforcing the starting point for these approaches as the RFID layer's feature partition as seen in the reference model.

As some tags have limited capability to use encryption and also existing intrusion detection or plausibility checking techniques can sometimes function poorly when supply chain visibility is low – perhaps as custodians fail to share data or RFID is not deployed widely – there can be custodian locations where plausibility checks have limited use.

A solution may be to use a *Synchronised Secrets* approach (Lehtonen et al. 2007c). In this approach, EPC Class-One Generation-Two tags store a random number that is changed every time the tag is read by a reader. Every time a tag is read, the back-end first verifies a tag's static identifier (which could be a cloned tag serial number). Clearly, this is a one-to-one (1:1) interaction between a tag and reader at the real world layer. If this number is valid, the back-end then compares the tag's synchronised secret to the one stored for that particular tag. If these numbers match, the tag passes the check, otherwise an alarm is triggered. After the check, the back-end generates a new synchronised secret that the reader writes on the tag. The plausibility check can determine if a tag has an outdated synchronised secret, in which case, either the tag is genuine but it has not been correctly updated (desynchronised) or someone has obtained and written an old secret to the genuine tag, or the genuine tag has been cloned and the cloned tag has been scanned by the legitimate reader. (Lehtonen et al. 2007c)

Although this check occurs at a single location, an anomaly arises if several tags with the same serial number have been read across the entire system (Lehtonen et al. 2007c). Thus, an outdated synchronised secret is evidence to suggest a tag cloning attack has occurred, but not which tag is the clone tag.

As the exchange of synchronised secrets is monitored by the back-end, a cloning attack could be identified to a time window and location window in a custodian's location. In effect, this is validating the sequential reading of tags at readers in the supply chain, which clearly shows that the underlying success of this approach depends on the organisation of readers in sequence at the real world layer and RFID layer. As the approach relies on associations between tags and readers, it has been modelled in the RFID layer's association space. Therefore, to show that such an approach could begin at the first custodian, it appears in the first custodian's framework in Figure 56.

In the Cardinal Health system the item-level tag synchronisation may have limited use early on in the supply chain. When individual items are on pallets, the read rate is between 7.8 percent and 14.3 percent; however, the inference process allows custodians to assume all the items are on a pallet (Bacheldor 2006a, b). Thus, it may not be possible to validate synchronised secrets until the tote level. However, at the same time, the attacker's ability to clone tags prior to this point by obtaining data from tags which are active may not be possible given the low read rate performance. Thus, the solution may have limited relevance at this stage in the pharmaceutical supply chain – evident as a result of having applied a 'whole of system' approach to analysis.

Finally, a problem may arise when in the last phase of a pharmaceutical supply chain a manufacturer sells a product to a retailer via a number of shipping agencies, and the retailer clones the tag and attaches it to counterfeits (Staake et al. 2005). Provided the counterfeiter does not update the database, nor does the customer register the deal, other customers may query the database, receiving a plausible history.

It is at this stage, the last custodian, that the concept of *secure authentication* could be applied as a solution. The *EPC Product Authentication Service* (Staake et al. 2005) uses cryptographic algorithms on tags to verify tag authenticity. A tag contains an identification number, a secret key, and a cryptographic support. Somewhere in the network is a device, a cryptographic unit (CU), responsible for authenticating the tag. Authentication occurs when a tag communicates its serial number to the CU whereby, the CU responds with a challenge back to the tag. The tag encrypts the message with its secret key and sends it back to the CU. The CU

verifies the response by checking it with the key stored in the database. An unauthorised tag would be detected when it fails to provide the correct response. (Staake et al. 2005)

Clearly in order for an EPC Product Authentication Service to operate, not only would the tag component need to support the required functionality, but the back-end would also need to support the management of secret keys, and this would need to be standardised across the entire system. As the approach relies on component functionality, it has been modelled in the RFID layer's component partition, even though it would also require elements from other places in the reference model.

For the Cardinal Health system (Bacheldor 2006a, b), such an approach may make sense to be deployed at the last phase of the pharmaceutical supply chain. It is at this point that tags on drugs are the most exposed as read rates of individual tags are highest at the pharmacy and customer level – having been disassembled from the layers of packaging. It is at this point that an attacker could easily gain physical access to a tag to obtain its serial number to be used in future tag cloning attacks, to facilitate future counterfeiting of products. However, it is at this point that a complete track and trace history of a tagged drug would have been established, meaning plausibility checks should have already verified whether counterfeits were in the system. Consequently, this may reduce the need to use encryption at this stage in the pharmaceutical supply chain.

10.2.4.1 SUMMARY

This section has used the reference model to organise some security solutions which could mitigate attacks in pharmaceutical supply chains, thereby, finalising a 'whole of system' approach to analysis of this specific case study. It was shown that a number of detection solutions would be required, at least for pharmaceutical supply chains, to address attacks across the entire system.

At the stages of deployment for these solutions, these solutions were shown to be taking advantage of certain underlying contexts in the domain space. For example, the Supply Chain Visualiser made use of the known location of custodians and physical constraints to determine what would constitute implausibility in an entity's movement. However, the way these solutions detect anomalies, shows that these

solutions do not exist at a single layer. Solutions require elements across the layers and partitions of the domain space, whether in one custodian location or across several custodian locations, to operate effectively.

Thus, a 'whole of system' approach to analysis, facilitated by the reference model, enables one to take into consideration; how a system's standard operations and attacks will influence one's recommendation that a system implement certain solutions. The outcome is a more practicable set of security requirements which take advantage of what the system has to offer, rather than what the system should be offering in order to support various solutions which are proposed.

In contrast to previous work (Rotter 2008; Mitrokotsa et al. 2010), which has localised security analysis, the suggestion in this chapter is that 'whole of system' analysis, once performed, leads to more effective security.

10.3 OVERALL SUMMARY

This chapter has used the relatively complex example of RFID in a pharmaceutical supply chain to illustrate the effectiveness of approaching security 'whole of system' by way of the reference model.

During analysis, using the model, crucial influences leading to the derivation of security requirements were considered. RFID has been used in pharmaceutical supply chains to form an electronic pedigree. To this end, in order to produce information which suitably creates a history of a product, RFID has to be integrated in the system. This integration involves not only RFID but also real world and strategic elements. The underlying system was characterised by: a variety of custodian locations which had implemented RFID in different ways and to different extents; limitations on tag reading at different stages; product assembly/disassembly complicating tag reads and also which entities were monitored; and when various identifiers in the system were enabled/purged. The reference model was illustrated as capable of capturing these elements.

The main analysis outcomes, having deliberated through the model, appear to be practicable security requirements. As the system model encapsulated the elements which influenced RFID implementation, threat analysis, using attack trees, indicated

that only a limited number of threats were relevant. On that basis, some solutions were suggested as effective when integrated in this context. Thus, the model enabled consideration of synergistic effects and interrelationships which influence security.

When the examples of previous work are considered (Rotter 2008; Mitrokotsa et al. 2010), which have taken a relatively localised view of security, it seems likely this alternative approach leads to a more effective understanding of security requirements.

Chapter 11

Conclusions and Further Work

11.1 CONCLUSIONS

This thesis began as an exploration of potential avenues for improving Radio Frequency Identification (RFID) security. The example problem used was that of tag cloning. In addressing this aim, the work considered RFID security using a 'whole of system' approach.

Thus, the principal focus of this thesis was the advantages that can be gained from applying a 'whole of system' approach to RFID. The major contribution made in this area was:

- A reference model which facilitates such a 'whole of system' approach. The structure of the model is made up of integrated layer and partition properties. When existing methods, which exhibit systematic qualities, were integrated into this reference model, the results suggested that greater insight into RFID security could be achieved when compared with existing models described by Rotter (2008) and Mitrokotsa et al. (2010).

Using this reference model, that facilitates a 'whole of system' approach to the analysis of security in RFID systems, five major contributions were made:

- To make possible a 'whole of system approach', a domain model was introduced that defines some of the fundamental properties of RFID systems. The model comprises a logical view: components, attributes, operations, and relationships. In addition, it included a data view: associations and features. These have resulted in a controlled vocabulary which was illustrated as suitable for the identification and description of RFID domain constructs.

- Results from building and validating a simulator, based on the domain model, suggested that the domain model is suitable for 'whole of system' analysis. Moreover, the simulator based on this model, was demonstrated as useful for solution analysis when applied to the problem of exposing clone tags.

- Attacks which were modelled as sequences over system layers formed an RFID attacker behaviour taxonomy. The taxonomy modelled attacks in two system types: authorisation and monitoring. It was illustrated that good

locations for solutions in these systems could be identified when this systematic approach was followed.

- Experimental work, which was suggested by the reference model, illustrated that a 'whole of system' approach to RFID security can make possible the identification of previously unexplored attack interception points. In this case, reprogramming an RFID reader via its application programming interface (API) to expose clone attacks in data. This suggests that security solutions can be found between the reader output and the filters in the middleware.

- Finally, the reference model was validated using the specific example of a pharmaceutical supply chain. The model was illustrated as capable of a 'whole of system' approach to security requirement elucidation. This also derived support for the model's potential use in more general cases as the system examined exhibited some generic system properties.

The following sections further expand upon the contributions of this thesis and describe in more detail the conclusions that can be drawn from them.

11.1.1 INDIVIDUAL METHOD CONSIDERATIONS

This section briefly considers how individual methods integrated into the reference model impart results across the whole model. These methods, which came from a variety of sources, when integrated into the reference model, made clear specific RFID information. This has an advantage when compared to previous work (Rotter 2008; Mitrokotsa et al. 2010), as it is possible to utilise any analysis method to the problem of security in RFID systems provided it is systematic in its approach.

In addressing specific analysis requirements, existing methods were applied over the layer and partition properties. Domain modelling approaches described the fundamental properties of RFID systems. Agent based modelling and general simulation principles demonstrated the feasibility of analysing systems through a domain model. The attack tree method organised attacks as sequences, over layers, to derive attacker behaviour taxonomy. Experimental work, demonstrated the benefits of beginning from a model, and then exploring the suggestions through a simulator. Using this process, analysis was improved, and thus, the finding that a

reader could be reprogrammed via an application programming interface (API), to detect attacks, was a relatively straightforward task. Using existing methods, integrated into the reference model's structure facilitates more effective systems analysis.

While the application of individual methods in the reference model validates the model's constituent partitions; it has been shown that the integration of these partitions over layers validates the 'whole of system' approach. This was demonstrated in the specific example of a pharmaceutical supply chain, when the whole model combined individual analysis methods. The ability to integrate individual analysis outcomes, discussed in previous chapters, but used here with the reference model structure for a complete and specific example, illustrates that the reference model does enable a 'whole of system' approach.

11.1.2 REFERENCE MODEL CONSIDERATIONS

This section briefly considers how the reference model makes possible a 'whole of system' approach by applying individual methods integrated using a layered and partitioned structure.

To facilitate a 'whole of system approach', the first step was to produce a domain model. The possibility of using layer and partition properties for this step was explored and illustrated to be of benefit by creating a logical view of the domain. This logical view defined the components, their attributes and operations. Conversely, a data view defined the associations and features which constituted the interactions which take place between components but which are reported in RFID data. Although the domain model is relatively modest, its level of abstraction demonstrates that it is sufficient to derive an understanding of any particular system. Further to this, the model is extensible, and therefore, future work may consider further domain analysis to achieve more detail. This contribution was reported by Mirowski et al. (2009c).

The task of showing the domain model to be suitable for representing systems at its level of abstraction was investigated from the perspective of Agent Based Modelling and Simulation (ABMS). A simulator that facilitates system and attack modelling was presented and was validated to be capable of predicting RFID data that is

sufficiently close to data from actual systems. Although it is a simplification of actual systems, it has been illustrated to be close enough to be useful in facilitating preliminary investigations in a 'whole of system' manner. The use of this simulator was illustrated to give impetus to experimentation with actual systems when results suggested by the simulator were examined in a laboratory. This contribution was reported by Mirowski et al. (2009c).

Taking a 'whole of system' approach to security means examining security throughout the layers of the system. Attacks, one part of security, were examined in the 'whole of system' approach using *attack trees* (Schneier 1999, 2004). Although attack trees were used, this thesis recommends any threat analysis method provided it can be applied systematically. Examining attacks from a systems perspective was illustrated to increase the understanding of where in a system it would be more beneficial to locate security solutions. This contribution was also reported by Mirowski et al. (2009b).

A scenario where many tags, some of which were clone tags, were in front of a reader was considered in a 'whole of system' approach. The finding was that the simple reprogramming of a reader can enable the exposure of clone attacks without requiring additional context. Provided an RFID system has the ability to be reprogrammed, it may situate security at these locations, where it is more cost-effective, and thus, prevent data from attacks entering the middleware.

The individual methods used for each part of this thesis are not specifically recommended for use; rather the notion that methods are integrated into layer and partition properties, to achieve integrated analysis, which can account for the synergistic effects and interrelationships in RFID, is what has been illustrated to be of benefit.

If the approach to RFID security requirements occurs 'whole of system', then the security understanding gained is likely to be improved, when compared to previous work (Rotter 2008; Mitrokotsa et al. 2010), and hence the security developed should be more effective.

11.1.3 GENERAL CONSIDERATIONS

Central to this work is a desire to emphasise the exploration of improving the security of RFID systems using a 'whole of system' approach. Generally, RFID security has been examined using localised approaches. The advantage of a 'whole of system' approach to RFID security requirements is that a wider view of a system can be taken, thus, more components can be incorporated into the security analysis. This thesis has illustrated that in the specific example of a pharmaceutical supply chain, a 'whole of system' approach leads to a more effective understanding of security requirements. As layer and partition properties are non-prescriptive in terms of the specific analysis methods to be used, and layer and partitions are properties applicable to most systems, it should be possible to apply the reference model to most systems.

Another key property of this work is the modular approach taken to reference model development. The methods which have been demonstrated as effective, over the model, can be replaced by other methods which could be applied across the layer and partition properties such that they are applicable to the modelling requirements which are under consideration at each partition. This modularity permits methods familiar to the user or most relevant to the modelling task to be selected.

Moreover, depending on the system under consideration, the number of layers and partitions can be varied. In some cases it could be more appropriate to analyse only one partition, whereas in other cases, additional layers or partitions may be needed to attain greater understanding. The modularity of the reference model allows it to be targeted to different problems.

11.2 FUTURE DIRECTIONS

Several major directions for future work arise from this research. Perhaps the most obvious is evaluating the generic applicability of the layered and partitioned reference model. Clearly this would be a major undertaking, as some systems, like supply chains, consist of different types of layer and partition properties at different stages. In undertaking this work, one could look towards the generic properties which form in systems, such as the associations between components.

Related to this, is the possibility of deriving more generic 'whole of system' models over the layer and partition properties. The attacker behaviour taxonomy was developed for attack goals which pertain to two specific system types; however, a complete enumeration of how attacks could be combined in systems was not explored. Future work here may consider identifying generic attack branches, and establishing reusable attack sequences for various system types.

In considering the completeness of the reference model, there are additional layer and partition properties which could be included in future versions. Fortunately the model follows an extensible design philosophy. There could be additional abstractions of the domain that encourage the addition of more partitions, and there could be enhancements to the ways in which RFID as a technology abstracts the real world. This may require the addition of further layers. One could look to including those layer or partition properties which would remain generic – like those included in the original version here – thereby leaving the minor properties which could vary between systems to system specific extensions.

Beyond this, another obvious avenue for improvement is to further automate methods for requirements analysis, perhaps by using the simulator that was developed. As it stands, the software can model simple system designs to visualise system activity and predict RFID data. Also, currently attack detection is a manual process, however, one which could be automated by utilising existing intrusion detection methods such as those put forward by Lehtonen et al.(2007c) or Mirowski and Hartnett (2007). In the future, it could be possible to automate the derivation of *attack signatures* for modelled systems, or the derivation of *trusted system* designs that take advantage of solution libraries accorded for the architecture of the underlying system. This would account for the nuances which arise as analysis moves closer to the actual application environment. In effect, the software platform could become an RFID system security prototyping tool.

Finally, the potential for architecture based security development could be further explored. This approach, which this thesis has taken the first steps in exploring, could be continued by others to expand the reference model to further enable the selection of security solutions in response to specific threats in different parts of an RFID system. Once a system has been modelled through the reference model, a

security pattern consisting of those solutions that are more effective in addressing threats given domain constraints, could be established. This reference model could become a general foundation for a 'whole of system' approach to RFID security analysis.

11.3 SUMMING UP

There is an abundance of existing research in RFID security which is applicable to the security of the components. Indeed, a significant amount of previous research exists on applying methods to securing RFID tag components. Rather than attempting to develop new methods to secure a particular component, the idea employed within this work has focused on a 'whole of system' approach to the analysis of security in RFID facilitated by a reference model.

This thesis has illustrated that by systematising the approach to RFID security through system layers, more effective analysis can be performed.

RFID technology is nearly always deployed with a 'whole of system' purpose in mind, and the ways in which systems are deployed are complex. As is often the case with complicated systems, various methods will exist for securing systems which will try to address various parts of a problem. If RFID technology history is anything to go by, it is nearly guaranteed that a single solution will not be developed that will resolve all potential threats in the near future; nor is it the case that a single system design will prevail such that it is possible to apply the same security strategy to all systems.

Thus, this thesis concludes with the point that practical results will be maximised by employing a range of individual analysis methods that are integrated through a 'whole of system' approach.

References

Agrawal, R., A. Cheung, K. Kailing and S. Schonauer (2006). Towards Traceability Across Sovereign, Distributed RFID Databases. Proceedings of the 10th International Database Engineering and Applications Symposium, pp. 174-184.

Alfred, R. (2008). "Dynamic Aggregation of Relational Attributes Based on Feature Construction " Advances in Databases and Information Systems **5207**, pp. 2-13.

AlienTechnology (2007). Reader Interface Guide: All Fixed Readers, September 2007, Alien Technology.

AlienTechnology (2008a). Alien Technology Java Developers Guide: September 2008, Alien Technology.

AlienTechnology (2008b). ALR-9650 Hardware Setup Guide, Alien Technology.

Arango, G. (1994). A Brief Introduction To Domain Analysis. Proceedings of the ACM Symposium on Applied Computing, pp. 42-46.

Avoine, G. and P. Oechslin (2005). "RFID Traceability: A Multilayer Problem " Financial Cryptography and Data Security **3570**, pp. 125-140.

Bacheldor, B. (2006a). "Cardinal Health Deems RFID Pilot a Success." Retrieved 25/01/2010, from http://www.rfidjournal.com/article/articleview/2838/.

Bacheldor, B. (2006b). "Cardinal Health Readies Item-Level Pilot." Retrieved 09/02/2010, from http://www.rfidjournal.com/article/view/2838.

Bacheldor, B. (2007). "Cardinal Health Deploying Drug E-Pedigree System." Retrieved 19/06/2009, from http://www.rfidjournal.com/article/view/3295/1/1/.

Bagui, S. and R. Earp (2003). Database design using entity-relationshp diagrams, Auerbach.

Balan, G. C., C. Cioffi-Revilla, S. Luke, L. Panait and S. Paus (2003). MASON: A Java Multi-Agent Simulation Library. Proceedings of the Conference on Challenges in Social Simulation.

Bapat, V. and G. Restivo (2005). "Reaping The Long-Term Benefits Of Integrating Radio Frequency Identification (RFID) Into Pharmaceutical Manufacturing." Pharmaceutical Engineering(3), pp. 32-44.

Benyon, D. (1997). Information and Data Modelling, McGraw Hill.

Bonabeau, E. (2002). Agent-Based Modeling: Methods And Techniques For Simulating Human Systems. Proceedings of the National Academy of Science of the United States of America, pp. 7280-7287.

Bono, S., M. Green, A. Stubblefield, A. Juels, A. Rubin and M. Szydlo (2005). Security Analysis Of A Cryptographically-Enabled RFID Device. Proceedings of the 14th USENIX Security Symposium, pp. 1-15.

Bruegge, B. and A. H. Dutoit (2004). Object-Oriented Software Engineering: Using UML, Patterns, and Java, Pearson Prentice Hall.

Chen, P. P.-S. (1976). "The Entity-Relationship Model - Toward A Unified View Of Data." ACM Transactions on Database Systems **1**(1), pp. 9-36.

Collins, J. (2004). "New Embeddable RFID Readers." Retrieved 18/07/2010, from http://www.rfidjournal.com/article/articleview/1114/1/1/.

Collins, J. (2006). "RFID-Zapper Shoots to Kill." Retrieved 10/09/2007, from http://www.rfidjournal.com/article/view/2098/1/1.

Courtois, N. T. (2009). The Dark Side Of Security By Obscurity And Cloning MiFare Classic Rail And Building Passes Anywhere, Anytime. Presented at the Workshop on RFID Security.

EPCglobal (2005). EPC™ Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz Version 1.0.9, EPCglobal.

FDA (2004). Combating counterfeit drugs: a report of the food and drug administration, U.S. Food and drug administration.

Fettke, P. and P. Loos (2003). Multiperspective Evaluation Of Reference Models – Towards A Framework Proceedings of the ER 2003 Workshops, Springer-Verlag, pp. 80-91.

Finkenzeller, K. (2004). RFID Handbook, Wiley.

Garcia, F. D., G. d. K. Gans, R. Muijrers, P. v. Rossum, R. Verdult, R. W. Schreur and B. Jacobs (2008). Dismantling MIFARE Classic. Proceedings of the 13th European Symposium On Research In Computer Security, Springer-Verlag, pp. 97-114.

Garfinkel, S. and H. Holtzman (2005). Understanding RFID Technology. RFID: Applications, Security, and Privacy. S. Garfinkel and B. Rosenberg, Addison Wesley**:** 15-36.

Garfinkel, S. and B. Rosenberg (2005). RFID: Applications, Security, and Privacy, Addison Wesley.

Gilbert, N. and S. Bankes (2002). "Platforms And Methods For Agent-Based Modeling." Proceedings of the National Academy of Science of the United States of America **99**(3), pp. 7197-7198.

Glover, B. and H. Bhatt (2006). RFID Essentials, O'Reilly.

Grasso, A. R. and P. H. Cole (2006). Definition Of Terms Used By The Auto-ID Labs In The Anti-Counterfeiting White Paper Series, Auto-ID Labs White Paper

Halamka, J., A. Juels, A. Stubblefield and J. Westhues (2006). "The Security Implications Of VeriChip Cloning." Journal of the American Medical Informatics Association **13**(5), pp. 601-607.

Hancke, G. (2005). A Practical Relay Attack On ISO 14443 Proximity Cards: Technical Report, University of Cambridge.

Hancke, G. P. (2006). Practical Attacks on Proximity Identification Systems (Short Paper). Proceedings of the IEEE Symposium on Security and Privacy, pp. 328-333.

Hashemi, K. (2009). Faraday Enclosures: Technical Report, Open Source Instruments Inc.

Hassan, T. and S. Chatterjee (2006). A Taxonomy For RFID. Proceedings of the 39th Hawaii International Conference on Systems Sciences, pp. 1-10.

Heydt-Benjamin, T., D. Bailey, K. Fu, A. Juels and T. O'Hare (2006). Vulnerabilities in First-Generation RFID-enabled Credit Cards, University of Massachuesetts.

Howard, M. and D. LeBlanc (2003). Writing Secure Code, Microsoft Press.

Howe, N., S. Goldner and C. Fennig (2007). "Drug Pedigrees: Your Supply Chain Needs Them. Are You Ready?" Pharmaceutical Engineering **27**(6), pp. 1-5.

Illic, A., T. Andersen and F. Michahelles (2009). EPCIS-Based Supply Chain Visualization Tool: Technical Report, Auto-ID Labs.

Inaba, T. (2008). EPC System For A Safe And Secure Supply Chain And How It Is Applied. Networked RFID Systems and Lightweight Cryptography. P. H. Cole and D. C. Ranasinghe, Springer**:** 191-210.

Juels, A. (2005). Strengthening EPC Tags Against Cloning. Proceedings of the 4th ACM Workshop on Wireless Security, pp. 67-76.

Juels, A. (2006). "RFID Security and Privacy: A research survey." IEEE Journal on Selected Areas in Communications **24**(2), pp. 381-394.

Juels, A., R. L. Rivest and M. Szydlo (2003). The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. Proceedings of the 10th ACM conference on computer and communications security, pp. 103-111.

Kasper, T., M. Silbermann and C. Paar (2010). All You Can Eat or Breaking a Real-World Contactless Payment System. Financial Cryptography and Data Security **6052**, pp. 343-350.

Kfir, Z. and A. Wool (2005). Picking Virtual Pockets Using Relay Attacks On Contactless Smartcard Systems. Proceedings of the 1st International Conference on Security and Privacy, pp. 47-48.

Kirschenbaum, I. and A. Wool (2006). How To Build A Low-Cost, Extended Range RFID Skimmer. Proceedings of the 15th USENIX Security Symposium, pp. 43–57.

Koh, R., E. W. Schuster, I. Chackrabarti and A. Bellman (2003). Securing The Pharmaceutical Supply Chain: White Paper, Auto-ID Center.

Korth, A. (2006). Modeling And Simulation With Agents. Computer Science Seminar, ACM.

Landt, J. (2005). "The History of RFID." IEEE Potentials **24**(4), pp. 8-11.

Law, A. M. (2005). How To Build Valid And Credible Simulation Models. Proceedings of the Winter Simulation Conference, pp. 24 - 32.

Lehtonen, M., F. Michahelles and E. Fleisch (2007a). Probabilistic Approach for Location-Based Authentication. Proceedings of the 1st International Workshop on Security for Spontaneous Interaction.

Lehtonen, M., F. Michahelles and E. Fleisch (2009). How to Detect Cloned Tags in a Reliable Way from Incomplete RFID Traces. Proceedings of the IEEE International Conference on RFID.

Lehtonen, M., N. Oertel and H. Vogt (2007b). Features, Identity, Tracing, and Cryptography for Product Authentication. Proceedings of the 13th International Conference on Concurrent Enterprising, France, pp.

Lehtonen, M., D. Ostojic, A. Ilic and F. Michahelles (2007c). "Securing RFID Systems by Detecting Tag Cloning." Pervasive Computing **5538**, pp. 291-308.

Luke, S., C. Cioffi-Revilla, L. Panait and K. Sullivan (2004). MASON: A New Multi-Agent Simulation Toolkit. Proceedings of the Eighth Annual Swarm User/Research Conference.

Macal, C. M. and M. J. North (2005). Tutorial On Agent-Based Modeling And Simulation. Proceedings of the Winter Simulation Conference, pp. 2-7.

Maciaszek, L. A. and B. L. Liong (2005). Practical Software Engineering, Pearson Addison Wesley.

Michael, K. and L. McCathie (2005). The Pros And Cons Of RFID In Supply Chain Management. Proceedings of the International Conference on Mobile Business, pp. 623-629.

Mirowski, L. and J. Hartnett (2007). "Deckard: A System to Detect Change of RFID Tag Ownership." International Journal of Computer Science and Network Security **7**(7), pp. 89-98.

Mirowski, L., J. Hartnett and R. Williams (2009a). How RFID Attacks are Expressed in Output Data. Proceedings of the 10th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN 2009), pp. 794-799.

Mirowski, L., J. Hartnett and R. Williams (2009b). "An RFID Attacker Behavior Taxonomy." IEEE Pervasive Computing **8**(4), pp. 79-84.

Mirowski, L., J. Hartnett and R. Williams (2009c). Tyrell: an RFID Simulation Platform. Proceedings of the Fifth International Conference on Intelligent Sensors, Sensor Networks and Information Processing, pp. 325-330.

Mirowski, L., J. Hartnett, R. Williams and T. Gray (2008). A RFID Proximity Card Data Set: Technical Report, School of Computing and Information Systems, University of Tasmania.

Mirowski, L. T. (2006). Detecting Clone Radio Frequency Identification Tags: Thesis. School of Computing. Hobart, School of Computing, University of Tasmania.

Mišic, V. B. and J. L. Zhao (2000). Evaluating The Quality Of Reference Models. Proceedings of the 19th International Conference on Conceptual Modeling pp. 484-498.

Mitrokotsa, A., M. Beye and P. Peris-Lopez (2010). Threats to Networked RFID Systems. Unique Radio Innovation for the 21st Century: Building Scalable and Global RFID Networks. D. Ranasinghe, M. Sheng and S. Zeadally, Springer-Verlag**:** 39-63.

Mitrokotsa, A., M. Rieback and A. Tanenbaum (2008). Classification Of RFID Attacks. Proceedings of the 2nd International Workshop on RFID Technology, pp. 73-86.

Mitrokotsa, A., M. Rieback and A. Tanenbaum (2009). "Classifying RFID Attacks and Defenses." Information System Frontiers **12**(5), pp. 491-505.

Nohl, K., D. Evans, Starbug and H. Plötz (2008). Reverse-Engineering A Cryptographic RFID Tag. Proceedings of the 17th USENIX Security Symposium.

O'Connor, M. C. (2005a). "French NFC Payment Trial Kicks Off." Retrieved 18/08/2009, from http://www.rfidjournal.com/article/articleview/1943/1/1.

O'Connor, M. C. (2005b). "Smart Packaging Sells Forklift Readers." Retrieved 18/08/2009, from http://www.rfidjournal.com/article/articleview/1465/1/1.

O'Connor, M. C. (2006). "GlaxoSmithKline Tests RFID on HIV Drug." Retrieved 22/09/2010, from http://rfidjournal.com/article/articleview/2219/1/1.

O'Connor, M. C. (2007). "Purdue Moving OxyContin RFID Pilot to Full Production." Retrieved 20/10/2010, from http://www.rfidjournal.com/article/view/3043.

O'Connor, M. C. and M. Roberti. (2005). "Impinj Announces Gen 2 Tags, Reader." Retrieved 05/07/2008, from http://www.rfidjournal.com/article/view/1482/1/1.

Oren, Y. and A. Shamir (2007). "Remote Password Extraction from RFID Tags." IEEE Transactions on Computers **56**(9), pp. 1292-1296.

Peris-Lopez, P., J. C. Hernandez-Castro, J. M.Estevez-Tapiador and A. Ribagorda (2006). "RFID Systems: A Survey On Security Threats And Proposed Solutions " Personal Wireless Communications **4217**, pp. 159-170.

Pressman, R. S. (2000). Software Engineering: A Practitioner's Approach, McGraw Hill.

Prieto-Diaz, R. (1987). Domain Analysis For Reusability. Proceedings of the 11th Computer Software & Applications Conference, pp. 23-29.

Ranasinghe, D. C. and P. H. Cole (2008). Networked RFID Systems. Networked RFID Systems and Lightweight Cryptography. P. H. Cole and D. C. Ranasinghe, Springer. **1:** 45-58.

Ranasinghe, D. C., D. W. Engels and P. H. Cole (2004). Low-Cost RFID Systems: Confronting Security And Privacy. Proceedings of the Auto-ID Labs Research Workshop.

Ranasinghe, D. C., M. Harrison and P. H. Cole (2008). EPC Network Architecture. Networked RFID Systems and Lightweight Cryptography. P. H. Cole and D. C. Ranasinghe, Springer**:** 59-78.

Reid, J., J. G. Nieto and T. Tang (2007). Detecting Relay Attacks with Timing-Based Protocols. Proceedings of the 2nd ACM symposium on Information, Computer and Sommunications Security ACM, pp. 204-213.

Rieback, M., B. Crispo and A. Tanenbaum (2006). Is Your Cat Infected With A Computer Virus? Fourth Annual IEEE International Conference on Pervasive Computing and Communications**:** 169-179.

Roberti, M. (2004). "EPCglobal Ratifies Gen 2 Standard." Retrieved 15/10/2010, from http://www.rfidjournal.com/article/articleview/1293/1/1/.

Robinson, S. (2004). Simulation: The Practice of Model Development and Use, Wiley.

Rotter, P. (2008). "A Framework For Assessing RFID System Security And Privacy Risks." IEEE Pervasive Computing **7**(2), pp. 70-77.

Sarma, S. (2001). Towards The 5-cent Tag: Technical Report, MIT Auto-ID Center.

Sarma, S. E., S. A. Weis and D. W. Engels (2003). RFID Systems And Security And Privacy Implications Proceedings of the Cryptographic Hardware and Embedded Systems, pp. 454-469.

Schneier, B. (1999). "Attack Trees." from http://www.schneier.com/paper-attacktrees-ddj-ft.html.

Schneier, B. (2004). Secrets and Lies, Wiley.

Shepard, S. (2005). RFID: Radio Frequency Identification, McGraw Hill Networking.

Spiekermann, S. and H. Ziekow (2005). RFID: A 7-Point Plan To Ensure Privacy. Proceedings of the 13th European Conference on Information Systems, 2005.

Staake, T., F. Thiesse and E. Fleisch (2005). Extending The EPC Network - The Potential Of RFID In Anti-Counterfeiting. Proceedings of the 2005 ACM Symposium on Applied Computing, pp. 1607-1612.

Swedberg, C. (2004). "RFID Drives Highway Traffic Reports." Retrieved 15/11/2010, from http://www.rfidjournal.com/article/articleview/1243/1/1.

Swedberg, C. (2010a). "High Demand Keeps Tag Prices Steady." Retrieved 15/10/2010, from http://www.rfidjournal.com/article/view/7901.

Swedberg, C. (2010b). "Sales of EPC RFID Tags, ICs Reach Record Levels." Retrieved 19/10/2010, from www.rfidjournal.com/article/articleview/7952/.

Tan, P.N., M. Steinbach and V. Kumar (2006). Introduction To Data Mining, Addison Wesley.

Teorey, T., S. Lighstone and T. Nadeau (2006). Database Modelling And Design, Morgan Kaufmann.

Thompson, D. R., N. Chaudhry and C. W. Thompson (2006). RFID Security Threat Model. Conference on Applied Research in Information Technology.

Wasserman, E. (2005). "Purdue Pharma To Run Pedigree Pilot." Retrieved 15/10/2010, from http://www.rfidjournal.com/article/view/1626/1/1.

Weinstein, R. (2005). "RFID: A Technical Overview And Its Application To The Enterprise." IEEE Computer Society **7**(3), pp. 27-33.

Wessel, R. (2007). "DHL to Market RFID-Enabled Smart Box." Retrieved 25/11/2010, from http://www.rfidjournal.com/article/articleview/2945.

Westhues, J. (2005). Hacking The Prox Card. RFID: Applications, Security, and Privacy. S. Garfinkel and B. Rosenberg, Addison Wesley**:** 291-301.

Wnek, J. and R. S. Michalski (1994). "Hypothesis-Driven Constructive Induction In AQ17-HCI: A Method And Experiments " Machine Learning **14**(2), pp. 139-168.

Zand, M. K. (1998). An Introduction to Software Reuse. Encyclopedia of Computer Science and Technology. A. Kent and J. G. Williams, Marcel Dekker**:** 73-88.

Zetter, K. (2007). "Scan This Guy's E-Passport And Watch Your System Crash." Retrieved 10/11/2007, from http://www.wired.com/politics/security/news/2007/08/epassport.

Zimmermann, H. (1980). "OSI Reference Model - The ISO Model Of Architecture For Open Systems Interconnection." IEEE Transactions on Communications **28**(4), pp. 425-432.

Appendix A

APPENDIX A - RFID SIMULATOR

This appendix is partially based on a publication presented at the 5th International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Melbourne, Australia, 2009 (Mirowski et al. 2009c)

A.1 OVERVIEW

This appendix reports on the development of a Radio Frequency Identification (RFID) simulator for the purpose of modelling systems, visualising system activity, and predicting output data. It addresses an outstanding research gap for RFID security; the problem of preliminary systems analysis without needing to examine actual system deployments.

To this end, the development process is explained, beginning with an overview of the motivation for developing the RFID simulator. Following this, the four phases of development are explained, and the results which report the simulator as validated for the purpose of preliminary 'whole of system' analysis are presented. The last phase of development, phase four, has already been reported in Chapter 8.

A copy of the Java program, which is the simulator reported in this chapter, can be found on the CD which accompanies this thesis.

A.2 MOTIVATION FOR DEVELOPING AN RFID SIMULATOR

The motivation for developing the simulator stems from a lack of attack evidence for researchers building intrusion detection systems. To this end, it was developed with the purpose of generating data containing attacks, and it also models generic systems for a variety of general system modelling problems.

As the RFID output data reports when events have occurred in the real world, research has looked for evidence of behavioural implausibility to identify attacks in the output data (Lehtonen et al. 2007a; Mirowski and Hartnett 2007). However, the problem has been that output data containing traces of attacks has not been available to researchers. This is necessary, as the attacks themselves are characterised by the RFID system in which they have occurred, and therefore, attack behaviour must be defined within a system's context for effective countermeasures to be developed.

The lack of available output data, to be used to develop countermeasures to detect clone tag attacks, was reported by Mirowski and Hartnett (2007). Using output data from a real proximity card RFID system, we synthesized the effects of tag cloning attacks. There were shortcomings with the approach: it was not known whether

attacks existed in the output data as it was obtained directly from a real system; the ability to realistically represent attacks relied on assumptions made on specific application knowledge; the data had to be sanitised to protect user privacy; and unpredictable behaviour of the entities made the system error prone. Since releasing the attack free output data on the internet (Mirowski et al. 2008), we have observed that it has had over 900 downloads[4]. This suggests a clear demand from researchers for RFID data.

Consequently, one began to identify reasons which further supported the need for an RFID simulator. A software based approach may avoid the time associated with generating output data when the activity of the entities which are producing the output data is minimal. For example, in a real system, entities could remain stationary in one place and therefore produce no output data as they are not in range of a reader. Also, simulation would allow for very large systems to be built, which contain many components and at little cost as hardware is only simulated. Thus, there is a time and cost saving by using a simulator.

A.3 DEVELOPMENT

The development process for the software was based on a four phase lifecycle, illustrated in Figure 57. The four phases of the development lifecycle were: conceptual modelling; implementation in software; verification and validation; and exploring the solution space. During each phase, learning outcomes which contributed to understanding about systems emerged, and these were fed back into the development cycle. After a number of iterations were performed, the software was deemed acceptable for use.

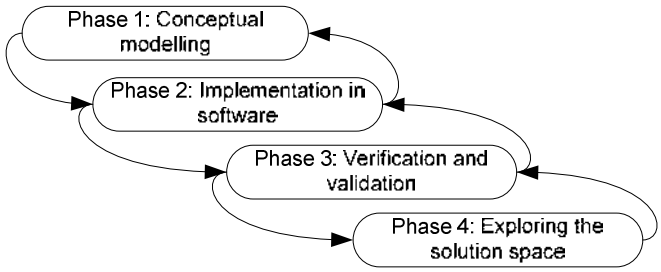

Figure 57 - Development lifecycle of the simulator

[4] Statistics were viewed on 16/12/2010 from the University of Tasmania electronic prints database at http://eprints.utas.edu.au/es/index.php?action=show_detail_eprint;id=6903;

The development of the simulator occurred in four phases, each of which provided the opportunity for feedback to a previous phase. The first phase was conceptual modelling which resulted in the encoding of standard components of RFID systems. The outcome of conceptual modelling was a domain model which was reported in Chapter 6. Once an initial domain model had been formed, the components were implemented in software as agents. This involved translating a high level Unified Modelling Language (UML) (Bruegge and Dutoit 2004; Maciaszek and Liong 2005) representation into a lower level software based implementation. It was through this translation process that concepts modelled in the conceptual model needed to be revised – hence some stages were repeated.

An example of a learning outcome that emerged during the feedback between cycles was that, the conceptual model had assumed that it would be necessary to model anti-collision at the algorithm level however this was found not to be the case. During implementation in software, it was learnt that anti-collision could be simplified as a single attribute and this was called *tag read rate*. This attribute contained a value which signified the number of data records a reader would produce within a time period for each tag instance at a reader. This avoided modelling the underlying anti-collision algorithm which essentially results in RFID data being produced once tags have been scheduled.

While simplifying a highly complex process, it was determined that such a simplification was acceptable at the level at which a 'whole of system' approach to RFID security was to be applied in this thesis. This meant that instead of thinking in terms of the algorithmic anti-collision layer e.g. slots, scheduling, and count-down timers, it would be possible to think in terms of the outcomes after anti-collision e.g. time between successive data records, total number of data records produced by a tag. These simplified complex processes, but reflected what was essentially achieved through their usage.

The rest of this section explains each development phase in the lifecycle.

A.3.1 PHASE ONE: CONCEPTUAL MODELLING

Conceptual modelling defined the standard components of the standard operating partition. The first focus was on enumerating the major components, their operations, and attributes. As a direct result of having to think about how these components could be simulated, the concepts of generic *components, associations* and *features* emerged. Whereas Chapter 6 reported on the completed domain model the focus here is on explaining how simulation development gave impetus for the domain model to emerge.

Components in RFID systems can be complex entities. Radio frequency, backscatter, and anti-collision are just a few of these concepts (Glover and Bhatt 2006). Furthermore, physical entities introduce concepts which are difficult to define in a static context as they only emerge when entities interact. For example, the effect of several tags interacting with a reader may result in a different effect at a reader than if a single tag was interacting. Therefore, some simplifications needed to take place.

A physical entity, for example, has a number of property simplifications in its conceptual model representation. Some of these are represented as attributes: s*peed_minimum*, *speed_average*, and *speed_maximum*. In the conceptual model, and simulator these are simplifications, whereas, in actual systems these would vary. Speed for example, is usually not a fixed value; however, to convey the concept to a non-expert that an entity can bring a tag into contact with a reader, it was assumed acceptable to simplify this as a static attribute.

Radio frequency signals were represented as a spatial concept, which is a simplification of the radio signals which exist in a real system. On the basis that in a real system, radio frequency signals are a factor in determining the range over which components can interact, these were modelled as simple three-dimensional spheres. For passive tags, typically, the frequency and power of a reader will determine the distance a tag can be from a reader for it to receive enough power to switch on, and also, the distance to send or receive data. Rather than simulate these complex processes, if a component is outside of a reader's signal then no interaction will be recorded; there is a cut-off point in order for data to be generated. Therefore, the

'final effect', which is the contact made between components, in the simulation, conveys the associations, and that is what is modelled.

Attacks were represented as operations performed by components. To specify an attack, the particular attack operation was instantiated by the component when it interacts with another component. To represent tag cloning attacks, a tag is assigned two new attributes: *public_name* and *private_name*. The public_name is how the duplicate *tag_serial_number* is specified and is the name which is recorded in the official data set, much like that in a real system. Conversely, the *private name* is the name used by the simulator or conceptual model to identify the clone tag from other tags. In real systems there is obviously only one data field, the public name field. In modelling tag denial of service, tags can use the following operations to model the effects of such an attack: *switch_on* or *switch_off* or *fail_to_respond*. As the attacks are modifying standard operations; the occurrence of these attacks can therefore be seen in the output RFID data which is produced when components interact. That is, issuing a *Kill* command to a tag could be modelled using the *fail_to_respond* operation. Thus, these higher level abstractions can achieve the same effects which would arise had these attacks occurred at a lower layer.

Anti-collision protocols have been represented by allowing tags within the vicinity of a reader to communicate with a reader, and hence, produce data at a reader. In a real system, anti-collision protocols are used to ensure that every tag is scheduled to respond, resulting in data being produced at a reader. However, it was assumed in this section that, as tags have to be scheduled in real systems, one factor which could occur is a variation in the data rate of a reader, and hence, for individual tags. That is, as more tags are scheduled to be read, a reader takes longer to service each tag request, thus, in a period of time, less data could be produced than when a single tag is read by a reader. As a way of modelling this, a *tag_read_rate* attribute was used to specify a fixed number of tags that could be read in a time period. This simplifies the delay factor as it records a fixed data rate – this is optimistic, whereas in reality this would vary according to the number of tags.

The primary learning outcome from the conceptual modelling phase was the derivation of a simplification of concepts and a controlled vocabulary which could be used to describe complex RFID operations. The simplification of anti-collision

protocol effects on the production of RFID data means non-experts could speak in terms of data rate for fixed tag group sizes at readers as opposed to needing to understand how collisions could impact on data production. Conversely, the use of entity-relationship terms (1:1, 1:M, M:1, and M:M) to describe relationships in RFID data which emerge as a consequence of underlying component associations, such as those at the tag and reader layer, simplify the identification of structures in these systems.

These concepts are reported in the completed domain model reported in Chapter 6, and in that chapter, are illustrated to contribute to a 'whole of system' analysis approach.

A.3.2 PHASE TWO: IMPLEMENTATION IN SOFTWARE

The conceptual model was implemented as an agent based simulation, using software called MASON (Balan et al. 2003; Luke et al. 2004). MASON provided the basic agent libraries, visualisation and scheduling, which have been used to construct RFID components as agents and have these interact with each other over time. To enable the specialisation of MASON into a program capable of modelling RFID systems, and to learn more about RFID attacks during the development process, additional features were included, and these were achieved using the Java programming language.

Figure 58 depicts the software's six core feature modules: conceptual model, application programming interface (API), script, simulation engine, three-dimensional (3D) animation, and output data. These features allow a user to model a variety of scenarios rather than a fixed simulation scenario. The API represents a list of commands which the user selects from to instantiate components from the conceptual model (called the 'domain model' in Chapter 6). A script contains all of the agents and instructions for a simulation scenario. An RFID system is essentially encoded in the script. The simulator executes the script, and thereby, executes a simulation, displaying the output, in the form of data and animation on screen.

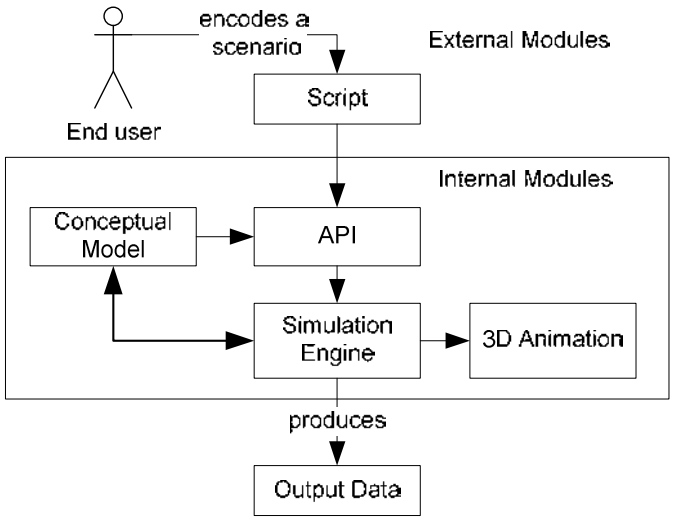

Figure 58 – Simulator's system boundary
A user engages the six modules when encoding a simulation. Various simulations can be modelled as the basis of modelling occurs from the application programming interface (API) – a library of commands which instantiate the domain model.

The rest of this section explains how these features are facilitative of 'whole of system' analysis.

Figure 59 depicts the software's graphical user interfaces (GUI) which allows the user to interact with the six core modules. In analysing RFID in a simulated environment, it was beneficial to provide a number of ways for a user to interact with the simulated system. The *expression builder* GUI is a visual tree representation of the API; commands can be selected to build up an API script to model a scenario. It is essentially an implementation of the controlled vocabulary which was derived through conceptual modelling. A script is formed in the expression builder's window or an already configured script can be loaded directly into this window. Execution of a script will cause the system to be built in the GUI. The user can visualise the system and the scenario as it is executed by the simulation engine. RFID *output data* – the data produced when tags and readers interact through the *associations* - is displayed in the log GUI. Thus, the GUI's are the means of instigating a 'whole of system' investigation in the software.

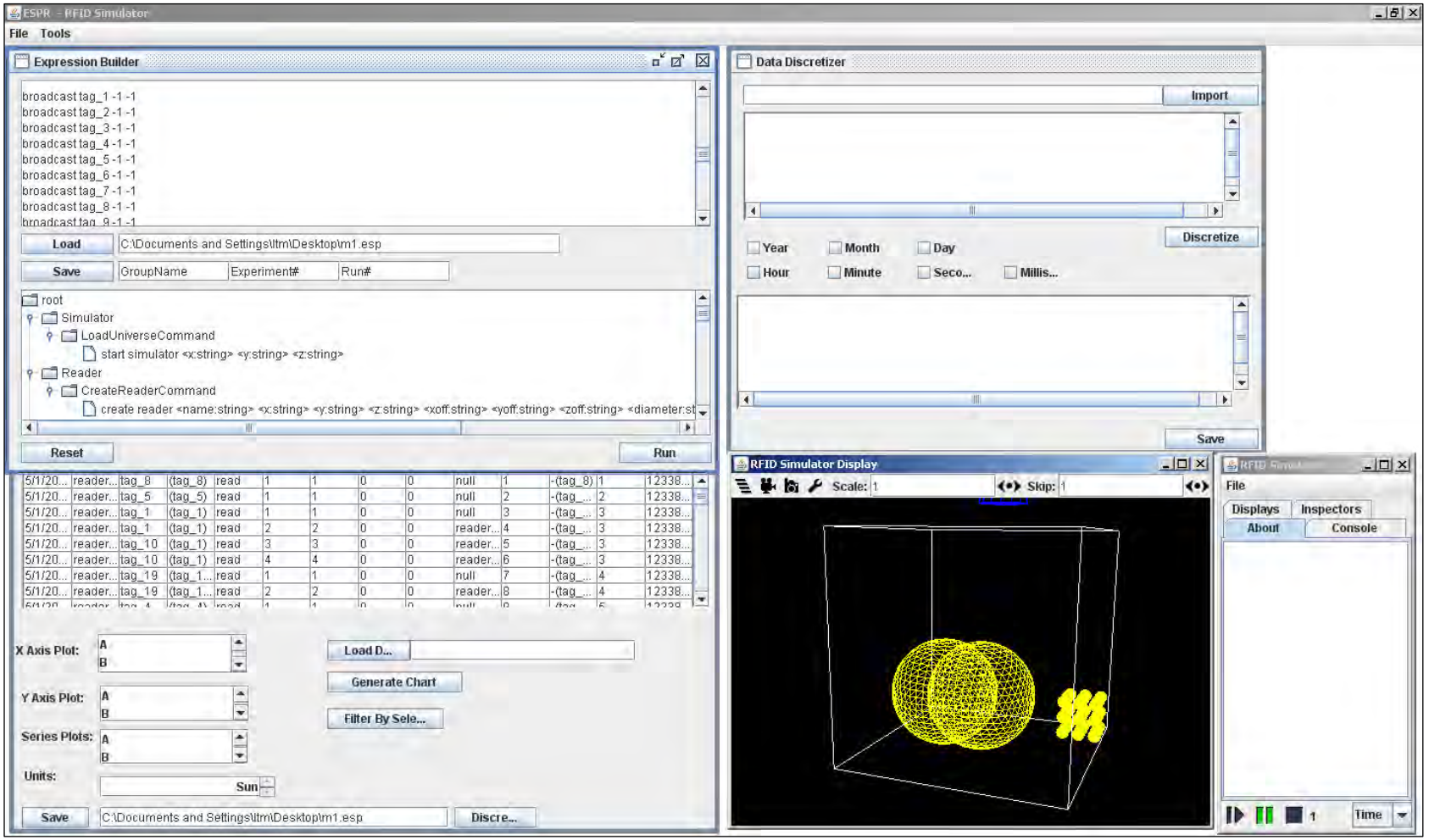

Figure 59 – Analysing RFID systems via a simulation model

The GUI allows the user to undertake a number of modelling tasks: encoding a scenario, executing it, and visualising the effect in data or in animation.

Table 8 depicts some of the commands from the API which are used to instantiate and instruct agents. An interface exists between the conceptual model and the API such that the issuing of a command will instigate a particular agent. A user specifies these commands inside a text file, called a *script*. A script is imported into the software and interpreted prior to execution. As a result of what appears in the script, particular instances of agents can be instantiated with specified values, queued in the schedule for execution, and so forth.

For a 'whole of system' approach, the outcome is that using this controlled vocabulary; users can communicate their understanding of the 'whole system' in a standardised manner. For example, it is possible to encode a system, insert attacks into it, and then use the script as the basis for determining how solutions could more effectively suit the given scenario. The commands standardise terms which could be used to describe the 'whole of system' approach.

Table 8 – Application Programming Interface (API)
Commands represent ways to instruct the simulation engine to load agent components and instigate interactions between them.

create physical entity <name:string> <x:string> <y:string> <z:string> <diameter:string>
create reader <name:string> <x:string> <y:string> <z:string> <xoff:string> <yoff:string> <zoff:string> <diameter:string> <database:string> <cloneName:string> <dataRate:string>
create tag <name:string> <x:string> <y:string> <z:string> <xoff:string> <yoff:string> <zoff:string> <diameter:string> <cloneName:string>
move <agentToMove:string> <stepsize:string> <x:string> <y:string> <z:string> <startTime:string> <intervalSize:string>
moveto <agentToMove:string> <stepsize:string> <agentToMoveTo:string> <startTime:string> <intervalSize:string>
read <reader:string> <startTime:string> <intervalSize:string>
broadcast <tag:string> <startTime:string> <intervalSize:string>

The API offers several benefits for 'whole of system' analysis. An RFID scenario can be encoded once and then replayed in the simulator many times, thus, promoting a repeatable method of producing output data containing attacks. As it is closely linked to the domain model, users can agree on the functionality expressed in a script, and on the output data produced by the execution of a script. Finally, the output data can be interpreted alongside the script to convey the context of the

system which would assist in strengthening an understanding of how structures influence attack detection.

Although the vocabulary is not complete, one could easily envisage that future work could look towards extending it such that a common way of modelling systems and security through a common dialogue exists. This could lead to common system descriptions, and common attack patterns, and consequently, common ways of dealing with attacks under various system contexts.

The simulation, when it is executing, is animated within a GUI in 3D facilitating learning via animation. Animation depicts what is happening inside the simulation engine at the current time period. Although this is not necessary for attack detection, when attacks are simulated, it enhances analysis as the interaction of agents can be seen by the user. For example, agents – tags and readers – are modelled as objects on the screen - in this version of the simulator they appear as 3D spheres. This depiction is an approximation of how radio frequency is typically depicted. It represents the geometric distance to which the signal propagates, and hence, can be detected by another component. It does not depict signal strength, which is something that deteriorates in an environment; thus, it is an optimistic representation of how far away a signal could be detected.

As this is a software implementation of components, the depictions can be modified to gain more precise representations if required. Hence, the software is extensible in this regard.

The physical entities which control physical manipulation of tags and readers are not modelled as agents in the simulator. At the moment, these remain external to the simulation as these were not of interest to the attack simulations this software was used for. The primary effect on tags, speed and movement is, however, inherited by these components in their place. A tag can therefore be instructed to move at a particular speed towards a location in the environment. Moreover, underlying associations between physical entities, for example, configuration of physical entities on a pallet (a M:1 association), are represented as associations between the tags that would be on the physical entities. This is because the goal is to model the structures, rather than the factors which lead to their creation.

Finally, analysis using the simulator can occur through the RFID output data which is produced when components interact. For this to occur, a tag and reader, need to be in the same location at the same time, and both be configured to interact. Interaction would result in output data being produced. An elementary data record consists of the following features: *tag_serial_number*, *reader_serial_number*, *reader_operation*, and *timestamp*. It is these records which are used by plausibility checks, however their derivation and interpretation depends on the structures. A benefit of providing the RFID output data in the simulation is that analysis can occur alongside interpretation of events in the simulation animation and the API script which is being executed.

In developing these features in the simulator, one was encouraged to think about the tools necessary for an analyst to take 'whole of system' approach to RFID security. The concept of the API, as a providing a controlled vocabulary for security proponents is one such example. The API's use as a basis for developing and communicating standardised approaches to security, allows different layers of a system to be encoded using command derived from the domain model.

A.3.3 PHASE THREE: VERIFICATION AND VALIDATION

As the simulator's goal was to enable thinking about a 'whole of system' approach to analysis, the verification and validation phase established confidence that the software was capable of modelling systems. This was established under the assumption that data should be produced by the simulator which contains evidence of attacks - tag cloning was the attack of interest here – and data should be similar to data from similarly built RFID systems.

The process of verification and validation was applied in the context of Mode Two simulations (Robinson 2004). The intentions of undertaking this validation were to establish whether: conceptual modelling and implementation in software had achieved a reasonable representation of systems; and that use of the simulator could result in reasonably accurate analysis outcomes from RFID data for 'whole of system' analysis.

Verification established confidence that the conceptual model was translated accurately into the software. The fact that various concepts have been included in

the depiction of agents using the Unified Modelling Language (UML) was a result of how systems were described. Determining that the conceptual model was transferred into the software occurred during the implementation phase for each agent. This involved a visual comparison between the Unified Modelling Language (UML) diagrams and Java classes which were developed. It was also established using the application programming interface (API) - by instantiating particular concepts through the API command set and by determining that the expected response could be produced in the software. Consequently, it appeared as though the software had captured the concepts in the conceptual model, and was therefore verified.

An initial validation process established confidence in the visual representation of RFID systems as agents. During implementation, a structured walk through with a group of users familiar with RFID systems, demonstrated representation of components as agents. The animation in the software was originally implemented in two dimensions (2D). However, based on user feedback, they found it easier to understand the ABMS analogy for RFID in three dimensions (3D). They could understand the concept of components as agents, the representation of radio frequency signal strength and signal propagation as geometric spheres. These users also accepted the characteristics of data as it was produced when components interacted under various structures. Thus, this initial stage developed *face validity* (Law 2005) in the simulator.

Next, a more complete form of validation established confidence in the software's ability to model RFID systems, and from these, that the data produced was valid for the purpose of its intended use. Validity in RFID output data was established by modelling a system in the software which also existed in the real world, and comparing the output data from both. This approach to validation is called *trace validation* (Robinson 2004).

A.3.3.1 TRACE VALIDATION

This section describes the validation of the simulator for preliminary 'whole of system' analysis under its mode of operation. The approach to 'whole of system' analysis relies upon the modelling of system context in which security analysis takes place. Based on the domain model, which contains simple system components, each of which contains simple operations, attributes, and relationships, the simulator

validation sought to establish the appropriateness of these elements in allowing a context to be modelled and a problem to be instantiated within it for analysis. The simulators mode was 'mode two' as a user (analyst) is expected to learn through usage of the simulator, in addition to the various data and animation outputs which result from experimentation. As simulation usage occurs prior to actual systems experimentation, the results obtained should be sufficient such that analysis of systems could proceed through experimentation and output such that information is gleaned which is on par with actual systems.

As a 'mode two' simulator, the user derives information through direct interaction with the simulator, as well as via various outputs. Conversely, the results are not expected to be an exact representation of an actual system, but be such that useful information is derived. The simulator is based on the domain model which has been instantiated in software and its use is facilitated via the simulator's application programming interface (API). The commands correlate to domain concepts which, when stored in a script, are instantiated when the script is executed. Through observation, the actual system is encoded within a script and these should be compatible such that animation and data, when produced in simulation or actual system are on par. Therefore, validation could occur on the basis of comparing the outcomes of a modelled system to those obtained in an actual system.

To this end, validation occurred using the specific example of a 'doorway monitoring system' which was implemented in the simulator and the results compared to a similar system implemented using actual RFID equipment. The doorway monitoring system was a reader mounted on a door which swings open or shut. A series of tags positioned directly under the door's pathway causes the reader, when it is within range of a tag, to produce a single data record at the tag. Each data record is stored in a database, and a series of data records, essentially, attests to a history of doorway activity over a period of time. Validation established: the ability of the API to encapsulate this scenario based on domain model concepts; animation of the interaction of scenario as similar to that of the real system; and the data derived out of the simulation as comparable to that of the actual system. These parts of the validation process are now discussed in more detail and will suggest that the simulator has been validated for the purpose of preliminary 'whole of system' investigations.

A.3.3.2 COMPARISON OF ENCODING TO ACTUAL SETUP

This section describes the comparisons made between the actual system and the encoding of the system using the API; both visually and on execution, and also conceptually.

Figure 60 - The doorway monitoring system

Figure 60 illustrates the actual system as implemented inside an office room with the reader mounted on the back of the door and tags positioned in the pathway of the reader as it opens or shuts. The doorway swings open or shut causing the reader to pass over the tags. Each tag contains a unique serial number, this, along with a timestamp is what is recorded in data. When a tag and reader interact, a data record is produced in the database which records this interaction. A series of records serves to indicate several pieces of information which attest to activity which has occurred in the doorway: opening/closing of doorway, speed of door, direction of opening/closing, and extent to which door was opened (based on which tags were read).

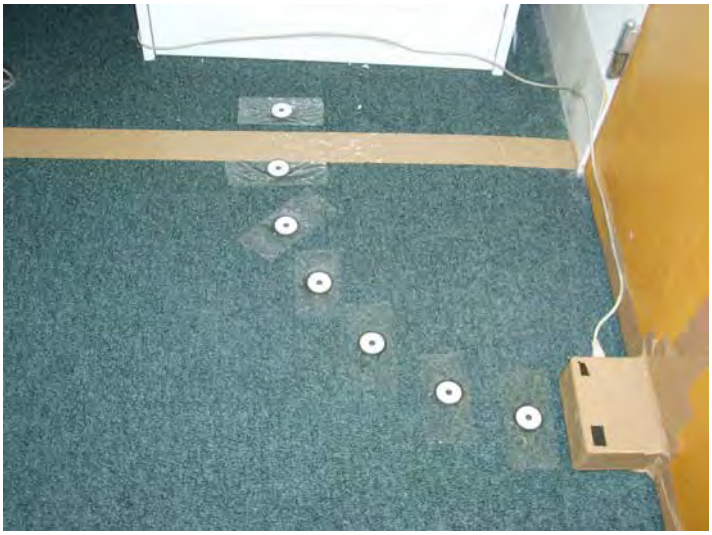

Figure 61 - The reader and tags mounted in the environment
When the reader passes over a tag, a data record is usually instantiated in the database. Sometimes interferences may occur such that not every tag is read e.g. the reader moves too quickly over the tags.

Figure 61 illustrates the tags taped onto the floor and with the reader mounted on the back of the doorway, in direct flight of the tags. The database is located on the computer connected via a cable to the reader; some of the elements such as the cable, for example, are tangential to the actual scenario.

What is now discussed is the comparison between the actual system and the modelled system for: visualisation and encoding using the API. Various illustrations provide evidence to suggest the simulator as capable of representing the system visually, whereas the API script illustrates the conceptual encoding of the system as similar.

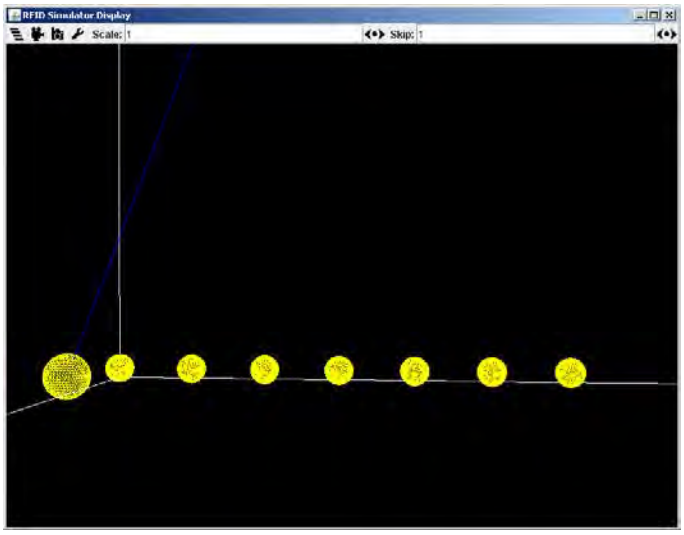

Figure 62 – Step 0 illustrates the reader at the start position of -2
The reader is the large sphere on the far left, with tags depicted as smaller spheres arranged in a straight line for the reader to pass over.

The application programming interface (API) was used to encode the same scenario, primarily those components which were of direct influence to the monitoring system: zone, door, tags and database. Figure 62 illustrates the organisation of tags and reader in a conceptual zone environment demarcated along three axes. The reader is the larger of the spheres, whereas the tags are the small spheres ordered in sequence. This captures the radio signal which emanates out of each of these components rather than their physical size in terms of form factor. An approximation of the organisation of tags was made in a 'line' rather than in an 'arc' for the purpose of simplifying the encoding using Cartesian coordinates. The ordering of tags as sequences was maintained with tag 01023c2807 the first tag (when the door is shut) and 01023bfb8e the last tag (when the door is fully open). Finally, the reader was instructed to move either forwards or backwards along the path of tags, whereas the tags are affixed to the zone and incapable of moving. This takes a similar appearance to the actual system.

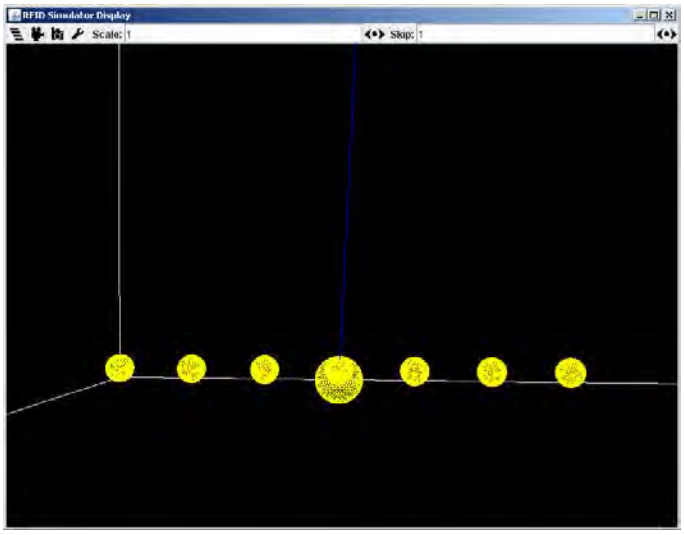

Figure 63 – Step illustrates the reader interacting with tag '01023c1baa'
As data is produced by the tag and reader interaction, the animation illustrates the same activity to the user on screen facilitating 'mode two' operation of the simulator.

Figure 63 and Figure 64 illustrate the movement of the reader across the tags in sequence as if the reader was affixed to the doorway. Rather than depict the doorway, using the domain models concept of 'speed', the reader was directly instructed to make this movement itself. The movement is relatively optimistic; in an actual system the doorway would arc across the tags, whereas in this example, the doorway (and hence reader) moves in a straight line. Another simplification is the fact that speed is applied at a constant rate; in an actual doorway, the doorway would progressively increase and decrease in speed upon opening and closing, whereas the simulator applies a single fixed speed. These are issues which have greater implications for the data which is produced when tags and readers interact, whereas visually, these appear to have minimal influence on the concept of component interaction.

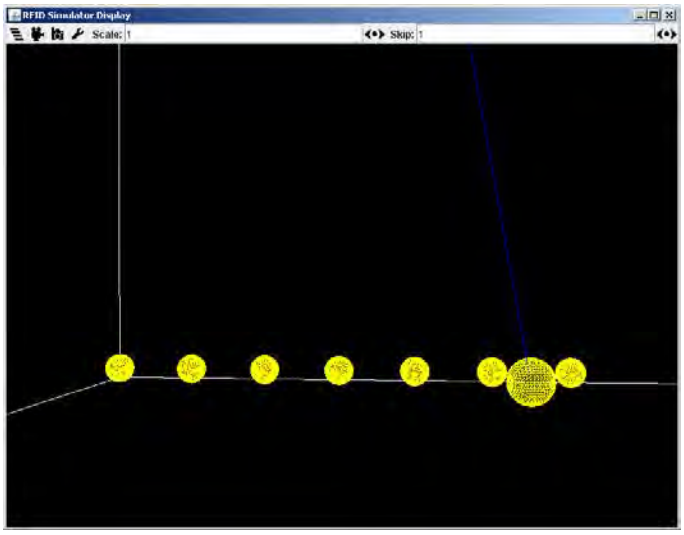

Figure 64 – Step illustrates the reader approaching the end of a 'run'
At this point, the simulator has produced six tags and its movement along the tags to this effect is illustrated on screen to the user.

The above figures have illustrated the representation of the actual system in the simulator's viewport as three dimensional objects. The encoding was a simplification of the layout, with tags placed in a line rather than an arc. The doorway was not physically instantiated; instead, the concept of the reader 'moving' across the tags, which is essentially the role of the door in this scenario, is encoded into the reader via its 'speed' attribute. When considering the above illustrations of the simulator to that of the actual system, it seems likely a close approximation exists in the visual representation of elements such that a user would find the depiction assistive in gaining information about the actual system via the animation.

Conversely, Figure 65 illustrates the encoding of the scenario using the API script using domain model concepts. Examination of the script reveals the conceptual similarities in what appears in the actual system when compared to the simulated system. The terminology in instantiating components and the issuance of instructions for components to move about the zone are apparent. The concept of validation here extends to one's ability to communicate the scenario using acceptable constructs such that the appropriate elements of the system are captured.

```
initialize{

setting TIME 16:40:06
setting DATE 05/01/2009

setting DISPLAY_X 800
setting DISPLAY_Y 600
setting TIME_UNITS Seconds
setting WORLD 1.0
setting PIXELS 10
setting UNITS CM
setting TICK_SCALE 1.0

start simulator 60 60 60
create tag 01023c2807 0 0 0 0 1 0 3 01023c2807
create tag 01023c15b5 8 0 0 0 1 0 3 01023c15b5
create tag 01023c364a 16 0 0 0 1 0 3 01023c364a
create tag 01023c1baa 24 0 0 0 1 0 3 01023c1baa
create tag 01023114f0 32 0 0 0 1 0 3 01023114f0
create tag 010230da56 40 0 0 0 1 0 3 010230da56
create tag 01023bfb8e 48 0 0 0 1 0 3 01023bfb8e

create database mydatabase 30 60 30
create reader phidget -6 0 0 0 0 0 5 mydatabase phidget 1

associate phidget mydatabase

}initialize


runtime{

read phidget -1 -1
move phidget 10 60 0 0 -1 -1

broadcast 01023c2807 -1 -1
broadcast 01023c15b5 -1 -1
broadcast 01023c364a -1 -1
broadcast 01023c1baa -1 -1
broadcast 01023114f0 -1 -1
broadcast 010230da56 -1 -1
broadcast 01023bfb8e -1 -1

}runtime


shutdown{
}shutdown
```

Figure 65 - The API script for the simulated 'doorway monitoring scenario

Figure 65 illustrates the API script which encapsulates the actual scenario using domain model concepts to be executed by the simulation. Initially, the script instructs the simulator to construct the physical zone for the system to be implemented inside, and nominally this was a 60 cubic centre-meters (cm) area which approximates the area in the actual system in which the system operates. The seven tags are instantiated in order of where they are to be positioned in the zone

using Cartesian coordinates with equal spacing between one another. Next, the reader and database are configured for data transmission and located in the simulation. The behaviour of the actual system is encoded in the last section; the reader is instructed to move across the series of tags at a fixed speed. Execution of the script instantiates the above scenario in the simulator, causing the system to be modelled on screen and execution ensures activity between tags and readers is visualised. When comparing the API script to the illustrations of the actual system, the domain model concepts facilitate encoding of the actual system, and on this basis, appear to be conceptually similar in organisation and execution.

To summarise, the encoding of the system in software was compared on the basis of visualisation in the viewport and using the API script. When considering the above examples, it seems likely that the encoded system has a similar representation as the layout of components appears similar. The execution of the script initiates the reader to pass over the tags, in a forwards or backwards motion, over the tags. This movement is on par with that experienced when the doorway is visualised in the actual system. By the same token, the API script attests to the encoding of the scenario; as domain concepts are used, a comparison of the encoding the actual system is evident in terms of what components exist in the environment, where these are located, and what actions these were instructed to undertake. It therefore seems likely that on this basis the simulator is valid in terms of its representative qualities.

A.3.3.3 COMPARISON OF DATA FROM SIMULATOR AND ACTUAL SYSTEM

This section discusses the comparisons made between the output data produced by the simulator and actual system. A single run of the simulator, using the above script, was compared to traces of runs from approximately 80 days worth of data collected from the actual system. The full data set can be found on the CD which accompanies this thesis. What follows is evidence to suggest that the simulator, even though it is based on a relatively simple domain model, can produce data which is 'on par' with that of an actual system.

Table 9 illustrates the output data obtained when the script was executed by the simulator. It can be seen that seven data records were produced within five seconds of the reader moving from its start position, hitting the first tag, and proceeding to its

end point which is beyond the final tag. The records, which appear in sequences as they were produced, indicate the reader travelled over the tags in sequence and in a forwards direction as if the door way swinging open.

Table 9 - Simulator output data
When executing the API script seven data records are produced which attest to the reader travelling across the tags in sequence.

Date & Time	Reader	Tag	Operation
5/1/2009,4:40:7:0	phidget	01023c2807	read
5/1/2009,4:40:7:0	phidget	01023c15b5	read
5/1/2009,4:40:8:0	phidget	01023c364a	read
5/1/2009,4:40:9:0	phidget	01023c1baa	read
5/1/2009,4:40:10:0	phidget	01023114f0	read
5/1/2009,4:40:11:0	phidget	010230da56	read
5/1/2009,4:40:11:0	phidget	01023bfb8e	read

In comparison, Table 10 illustrates the output data obtained for a run in the actual system which exhibits markedly similar characteristics. The records 710144 to 710151 constitute a complete 'run' which is when the door moved over all the tags, producing data at tags. It can be seen that these records were also produced within six seconds (one second more than the simulator) and in order of most of the tags which were sequenced over the floor. It can be seen that there are several inconsistencies: an extra data record was produced and several tags in the sequence are repeated even though the sequence starts and ends with the correct tags. It is this stochastic behaviour, at intermediate points in the scenario, which the simulator is unable to reproduce.

It should be noted that this table has been adapted marginally from the actual data set on the CD in that date & time columns have been rearranged and the reader column added – this way it has similar structure to the table above for easier comparison.

Table 10 – Actual system output data for run constituted by records adapted from records 710144 to 710151
This run indicates eight data records were produced in a similar time frame when compared to the above simulator example.

Date	Time	Reader	Tag	Operation
3/09/2008	15:4:24:546	phidget	01023c2807	gained
3/09/2008	15:4:26:859	phidget	01023c15b5	gained
3/09/2008	15:4:27:281	phidget	01023c364a	gained
3/09/2008	15:4:27:781	phidget	01023c1baa	gained
3/09/2008	15:4:28:250	phidget	01023114f0	gained
3/09/2008	15:4:28:343	phidget	01023c1baa	gained
3/09/2008	15:4:28:437	phidget	01023114f0	gained
3/09/2008	15:4:29:15	phidget	01023bfb8e	gained

Recall that the simulator is optimistic when it comes to modelling a scenario. Based on the domain model which has fixed data fields for such things as 'speed', the simulator does not account for the gradual increase and decrease which would be exhibited in an actual doorway system, when the door is being opened and shut. The simulator also does not model environmental interferences which occur in actual systems: the door is not fully opened; the door opens too quickly for all tags to be read; the door hits the doorstop and bounces back across some tags thereby producing 'runs' with noise in them. Nor does the simulator model the stochastic behaviour of the door-opener: different people open or close the door. Such conditions in the actual system would have influenced the production of data from something which is relatively predictable to something which is difficult to model.

Consequently, to provide some indication of the usefulness of fixed data fields when it is known that the values in these fields, in actual systems varies, a brief summary of observations of the data from the actual system data is listed:

- 307 complete runs were identified over 80 days of system activity. These are when the first and last tag was read within a time period constituting a full opening of the door.
- 287 of these runs produced 7 data records or less
- Only 10 runs produced exactly 7 records (the right number) whereas 248 runs produced 5 records. As it appears as though two tags were usually missed by

the reader and these were centre tags, the door may have been travelling too quickly for tags to be read.

When considering the above observations of the actual system data, it seems likely that the simulator is highly optimistic in terms of its data production for a modelled system. The data it produces represents the 'upper bound' on what one may expect the system to produce for the input settings. In the actual system, only 10 runs out of 307 produced exactly 7 records, whereas the simulator, for this scenario, would always produce exactly 7 records. Most of the time, the actual system produced 5 data records per complete run. The simulator's lack of variability in entity speed, for example, or consideration to environmental influences such as tag/reader interferences, may account for its optimism in data production. Conversely, if one required less optimistic data, it may then be appropriate to modify the domain model such that it could account for 'noise' in the system. Although this may not be appropriate for the simulator's existing intended usage as a mechanism for preliminary investigations. However, the questions of whether preliminary analysis would be likely facilitated on optimistic results as opposed to less than optimistic results, is a question which should be explored by further work.

Finally, the simulator, as simple as it is, will only ever account for the things which it has been instructed to model. In actual systems, there may be influences which are not apparent to the user, which therefore endear an exact representation in the simulation. The simulation works on the basis of the user observing the organisation and hardware of an actual system, or theorising systems RFID hardware settings and configuration, and simulating from that description. It does not encourage exactness in such things as: anti-collision, radio frequency, or stochastic entity behaviour. Its purpose is to be a mode two simulator which facilitates learning through a combination of simulation interaction as well as some simulation output. These facets, combined with user-driven analysis, should provide sufficient impetus and optimisation of search for solutions prior to actual systems experimentation.

The process of validation of the simulator, which is based on the domain model, was reported in this section. Validation examined its ability to encode the specific example of a doorway monitoring system using commands within the application

programming interface (API), and then execute this script within simulation to produce markedly similar representation and output data.

The results have suggested that a relatively simple system can be encapsulated although some of the limitations are apparent. Commands were able to encapsulate all relevant concepts for the purpose of modelling the system to produce output data. Tags and readers were represented as three-dimensional spheres in a zone. Conversely, commands in the API did not have the capacity to capture variability in entity attribute values for things like speed. This meant that data produced was optimistic in terms of the number of data records produced when compared to those collected in the actual system.

As a mode two simulator, the simulator appears as valid on the basis that information can be gleaned from it through user interaction via API; viewport/animation; and output data. Although some differences in output data would be obtained when compared to an actual system, when all facets of the simulator are considered, it appears suitable for preliminary 'whole of system' analysis.

A.3.4 PHASE FOUR: EXPLORING THE SOLUTION SPACE

Finally, while four phases are depicted in the development lifecycle diagram in Figure 57, the fourth phase is actually the outcomes of using the model – these are reported in Chapter 8.

A.4 SUMMARY

The outcomes from developing the simulator can be summarised as follows:

- A repeatable method for 'whole of system' analysis was developed which is based on the domain model. It enables preliminary 'whole of system' investigations to begin in software prior to analysis on actual systems.

- The concept of a controlled vocabulary, captured in the domain model, but implemented as an application programming interface (API), offers a way for analysts to communicate system designs.

- Verification and validation phases illustrated support for the usage of the simulator as a tool for 'whole of system' analysis.

Although the simulator is not a complex program, nor one that captures the complex interactions of RFID technology, its extensible design which is based on a domain model, facilitates future enhancements. This work has illustrated that the simulator's current instantiation is useful for preliminary 'whole of system' analysis expounded in this thesis.

Appendix B

APPENDIX B - SOURCE CODE FOR EXPERIMENTS

B.1 OVERVIEW

This appendix briefly describes the Java program written to perform the experiments with the Electronic Product Code (EPC) Class-One Generation-Two hardware. The source code can be found on the CD which accompanies this thesis.

The software has been written specifically for use with an Alien Technology EPC Class-One Generation-Two ALR-9650 reader. The software utilises commands from the reader's application programming interface (API) to instruct the reader into a particular operating state at various stages of experimentation. For the purposes of this thesis: the reader is reset prior to an iteration; a new configuration is written to the reader; the reader is instructed to poll the field for tags for a single Inventory Cycle; and the responses collected by the reader are obtained and written to file.

The software also behaves in accordance with the behaviour of the reader such that it captures the reader's data which is streamed back to the software. After issuing the inventory command to the reader, the software will wait a predefined amount of time before resetting the reader. This is because the reader does not indicate when it has obtained all tag responses; however, the assumption is that it does so within a few milliseconds. Thus, this software will only work for this reader; however, the general principles of experimentation and obtainment of data should be applicable to all EPC Class-One Generation-Two hardware.

All reference manuals for setup of hardware in addition to other API commands can be sourced from AlienTechnology (2007), AlienTechnology (2008a) and AlienTechnology (2008b).

Installation, usage guidelines, and other relevant information for the software can be found in the *Readme.txt* file under this appendix on the CD.

Appendix C

APPENDIX C - EXPERIMENT RESULTS

C.1 OVERVIEW

This appendix describes the full set of results obtained through experiments with Electronic Product Code (EPC) Class-One Generation-Two equipment. A brief description of the data is provided here, while all data sets can be found on the CD which accompanies this thesis.

```
Rep:1 of 15
[93][11]
[6][63]
[82][77]
[52][94]
[18][26]
Have you configured the tags? y/n
Issuing Command:get TagList
TAG STREAM 0
#Alien RFID Reader Tag Stream
#ReaderName: alien2
#Hostname: alien-00074C
#IPAddress: 10.1.1.10
#CommandPort: 23
#MACAddress: 00:1B:5F:00:07:4C
#Time: 2010/10/25 04:46:45.868
TAG STREAM 1
Tag:E200 3412 DC03 0117 5523 9646, Disc:2010/10/25 04:46:45.830,
Tag:E200 3412 DC03 0117 5523 9302, Disc:2010/10/25 04:46:45.840,
Tag:E200 3412 DC03 0117 5523 9198, Disc:2010/10/25 04:46:45.850,
TAG STREAM 2
Tag:E200 3412 DC03 0117 5523 9295, Disc:2010/10/25 04:46:45.980,
Tag:E200 3412 DC03 0117 5523 9471, Disc:2010/10/25 04:46:45.990,
Tag:E200 3412 DC03 0117 5523 9347, Disc:2010/10/25 04:46:45.990,
Tag:E200 3412 DC03 0117 5523 9211, Disc:2010/10/25 04:46:46.000,
Tag:E200 3412 DC03 0117 5523 9305, Disc:2010/10/25 04:46:46.000,
Tag:E200 3412 DC03 0117 5523 9281, Disc:2010/10/25 04:46:46.010,
Tag:E200 3412 DC03 0117 5523 9323, Disc:2010/10/25 04:46:46.020,
Finished Rep:1 of 15
```

Figure 66 - Sample of experiment results
The results depicted above illustrate the format of data and ordering of results which is found on the accompanying CD.

Figure 66 illustrates the format and ordering of results from an iteration which was performed during experimentation. The first part indicates the repetition number of the total number of repetitions. The second part indicates the physical tags that were randomly selected for use by the software, and also indicates their position in the Faraday enclosure. Once the user has confirmed that the physical tag setup is complete, the *get TagList* command is issued by the software – the result of this issuance is recorded in the results. The results are then collated in tag streams. A tag stream is the way the software organises collected results. When the population of tags is large, results for each inventory cycle will be recorded across multiple tag streams. Once all data has been collected by the software, as supplied by the reader, the software will terminate. This termination is indicated by the *Finished* statement on the last line.

The organisation of the individual data files which record the results of each experiment are reported in the *Readme.txt* file under this appendix on the CD.