

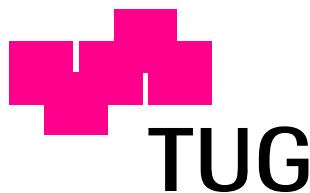
Magisterarbeit

# **An Evaluation of Security Threats and Countermeasures in Distributed RFID Infrastructures**

Stefan Stadlober Bakk. techn.

---

Institut für Informationssysteme and Computermedien  
Technische Universität Graz  
Vorstand: O. Univ.-Prof. Dipl.-Ing. Dr. techn. Hermann Maurer



Begutachter: O. Univ.-Prof. Dipl.-Ing. Dr. techn. Hermann Maurer  
Betreuer: Univ.-Ass. Dipl.-Ing. Dr. techn. Denis Helic

Graz, Juli 2005

## Zusammenfassung

Bereits in den 60er-Jahren des vorigen Jahrhunderts wurden die ersten kommerziellen Vorläufer der RFID Technologie auf den Markt gebracht. Dennoch ermöglichten erst die in letzter Zeit stark fallenden Preise und die voranschreitende Standardisierung den großflächigen Einsatz von RFID Systemen in der Wirtschaft. Diese Entwicklung wird von den Big Players der Retail Industrie zusätzlich vorangetrieben und bringt unsere Gesellschaft dem Ubiquitous Computing Szenario einen Schritt näher. Dieses Szenario zeichnet sich durch stark verteilte Datenverarbeitung, Just-in-Time Networking, und Location Awareness aus. Zusätzlich weisen RFID Daten einen engen Bezug zu Ort und Interaktionstyp auf. Durch die zunehmend automatisierte Verwaltung und Verarbeitung der RFID Daten wird es immer wichtiger, Sicherheitsmechanismen und Strategien zu definieren, um Sabotage, Missbrauch und Betrug zu verhindern. Diese Diplomarbeit, die in Zusammenarbeit mit Infineon Technologies Graz, Austria geschrieben wurde, untersucht Sicherheitsrisiken und Gegenmaßnahmen sowohl auf der Luftschnittstelle von RFID System als auch in verteilten RFID Middleware Architekturen. Die Risiken werden angeführt und analysiert und schließlich in Bezug zu den Kosten und der Effizienz von Gegenmaßnahmen gebracht. Derselbe Ansatz wird zur Analyse von Privacy Issues, die mit RFID verbunden sind, verwendet. Schließlich erstellt diese Arbeit einen Überblick über den gegenwärtigen RFID Markt, wirtschaftliche Aspekte und Zukunftsperspektiven.

## **Abstract**

The technology behind Radio Frequency Identification (RFID) has been around for a while, but dropping tag prices and standardization efforts are finally making the use of RFID for mainstream applications possible. Economies of scale associated with RFID and the market power of the US American and European big players in the retail industry will drive the adoption of RFID forward, taking us closer to the well-known ubiquitous computing scenario. Ubicomp as it is sometimes called, is characterized by heavily distributed computing, just-in-time networking and location awareness. This development along with the trend towards automated data management and processing will make it increasingly important to define cocksure security mechanisms to prevent fraud and abuse, which can have devastating effects in the context of ubiquitous computing. This diploma thesis, which was written in cooperation with Infineon Technologies Graz, Austria analyzes security threats as well as possible countermeasures on the air interface of RFID systems on one hand, and in distributed RFID middleware architectures on the other hand. The risks are assessed and relativized with the costs and efficiency of countermeasures. For the discussion of privacy issues associated with RFID the same approach is used. Furthermore this thesis gives an overview of the current RFID market, business aspects and future developments.

## Danksagung

*Mein herzlicher Dank gilt meinen Betreuern Denis Helic von der TU Graz und Andreas Kerschbaumer von Infineon für die ausgezeichnete und flexible Betreuung. Weiters möchte ich mich bei allen Personen bedanken, die mich bei der Durchführung dieser Arbeit unterstützt haben.*

Graz, im Juli 2005

STEFAN STADLOBER

# Contents

<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>vii</b>
<b>List of Abbreviations</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Security Primer</b>	<b>4</b>
2.1 Security Services . . . . .	4
2.1.1 Authentication . . . . .	5
2.1.2 Confidentiality . . . . .	5
2.1.3 Integrity . . . . .	5
2.1.4 Nonrepudiation . . . . .	5
2.1.5 Summary . . . . .	6
2.2 Symmetric Cryptography . . . . .	6
2.2.1 Block Ciphers . . . . .	7
2.2.2 Stream Ciphers . . . . .	7
2.2.3 Symmetric Algorithms . . . . .	7
2.2.4 Password Based Encryption . . . . .	7
2.3 Asymmetric Cryptography . . . . .	8
2.3.1 Asymmetric Algorithms . . . . .	8
2.4 Digital Signatures . . . . .	9
2.4.1 Digital Signature Algorithms . . . . .	10
2.5 Public Key Infrastructures . . . . .	10
2.6 Security Protocols . . . . .	10
2.6.1 Internet Protocol Security (IPSec) . . . . .	11
2.6.2 Secure Sockets Layer (SSL) . . . . .	12
2.6.3 Kerberos . . . . .	13
2.6.4 Web Services Security (WSS) . . . . .	15
2.7 Security Threats in Distributed Systems . . . . .	16
2.7.1 Anatomy of Attacks . . . . .	16
2.7.2 Classification of Attacks . . . . .	17
2.7.3 Network Threats . . . . .	18
2.8 Summary . . . . .	20
<b>3 Radio Frequency Identification</b>	<b>21</b>
3.1 From Barcodes To RFID . . . . .	21
3.2 Classification of RFID Systems . . . . .	22
3.2.1 Frequency, Range and Coupling . . . . .	22
3.2.2 Information Processing in the Transponder . . . . .	23

3.2.3	Other Classification Criteria . . . . .	24
3.3	RFID Transponders . . . . .	25
3.3.1	Transponders with Memory . . . . .	25
3.3.2	Transponders with Microprocessor . . . . .	27
3.4	RFID Readers . . . . .	28
3.5	Dataflow in RFID Systems . . . . .	28
3.6	RFID Standards . . . . .	29
3.6.1	ISO/IEC 14443 Proximity Cards . . . . .	30
3.6.2	ISO/IEC 15693 Vicinity Cards . . . . .	31
3.6.3	ISO/IEC 18000 . . . . .	32
3.6.4	Electronic Product Code (EPC) . . . . .	33
3.7	Proprietary Systems . . . . .	34
3.7.1	Infineon's My-d . . . . .	34
3.8	Summary . . . . .	35
<b>4</b>	<b>Security Aspects of RFID</b>	<b>36</b>
4.1	Security in the Context of RFID . . . . .	36
4.1.1	Security Services Revisited . . . . .	36
4.1.2	Fraud Scenarios in RFID Systems . . . . .	37
4.2	Current RFID Security Mechanisms . . . . .	40
4.2.1	Authentication of Tags and Readers . . . . .	40
4.2.2	Encryption . . . . .	45
4.3	Additional Considerations . . . . .	48
4.3.1	Key Management . . . . .	48
4.3.2	Diversified Keys . . . . .	49
4.3.3	Multi-Application Support . . . . .	50
4.4	Security Threats Evaluated . . . . .	51
4.5	Summary . . . . .	54
<b>5</b>	<b>RFID and Privacy</b>	<b>57</b>
5.1	Threats to RFID Privacy . . . . .	57
5.2	Privacy Scenarios . . . . .	58
5.2.1	Data Privacy . . . . .	58
5.2.2	Location Privacy . . . . .	58
5.3	Current Approaches to Guaranteeing RFID Privacy . . . . .	59
5.3.1	Tap-Proof Anti-collision Protocols . . . . .	59
5.3.2	Anonymization of Tags . . . . .	60
5.3.3	Disabling Access . . . . .	61
5.3.4	Permanent Deactivation . . . . .	61
5.4	Privacy Threats Evaluated . . . . .	62
5.5	Summary . . . . .	64
<b>6</b>	<b>RFID Applications</b>	<b>66</b>
6.1	Application Areas of RFID . . . . .	66
6.1.1	Access and Route Control . . . . .	66
6.1.2	Document Verification . . . . .	67
6.1.3	Asset Management . . . . .	67
6.1.4	Supply Chain . . . . .	68
6.2	Implementations . . . . .	68
6.2.1	The EPCglobal Network . . . . .	68
6.2.2	E-Government . . . . .	70
6.2.3	Near Field Communication (NFC) . . . . .	70

6.2.4	Machine Readable Travel Document (MRTD) . . . . .	71
6.3	RFID - Chances and Risks . . . . .	72
6.3.1	Economical Aspects . . . . .	72
6.3.2	Legal Regulations . . . . .	72
6.3.3	Technical Aspects . . . . .	73
6.3.4	Standardization . . . . .	73
6.3.5	Integration Costs . . . . .	73
6.3.6	Security . . . . .	73
6.3.7	Privacy . . . . .	74
6.4	Summary . . . . .	74
<b>7</b>	<b>Securing a Distributed RFID Infrastructure</b>	<b>77</b>
7.1	RFID Infrastructures . . . . .	77
7.1.1	Requirements . . . . .	77
7.1.2	Architecture . . . . .	79
7.2	Threat Model . . . . .	82
7.2.1	Identification of Assets to Protect . . . . .	82
7.2.2	System Model . . . . .	83
7.2.3	Identification of Entry Points . . . . .	84
7.3	Implementing Security . . . . .	87
7.3.1	Security Models . . . . .	87
7.3.2	Authentication and Authorization . . . . .	88
7.3.3	Secure Communication . . . . .	92
7.4	Summary . . . . .	94
<b>8</b>	<b>Conclusion</b>	<b>97</b>
8.1	Summary . . . . .	97
8.2	Outlook . . . . .	98
	<b>Bibliography</b>	<b>99</b>

# List of Figures

2.1	Network Security . . . . .	11
2.2	Transport Security . . . . .	13
2.3	The Kerberos Protocol . . . . .	14
2.4	Anatomy of Network Attacks . . . . .	16
3.1	RFID Tag with Memory . . . . .	26
3.2	RFID Tag with Microprocessor . . . . .	27
3.3	RFID Communication Model . . . . .	29
3.4	Overview of RFID Standards [KP04] . . . . .	30
4.1	Intentions behind Attacks on RFID Systems . . . . .	39
4.2	Challenge-Response Authentication . . . . .	43
4.3	Eavesdropping on Reader-Tag Communication . . . . .	46
4.4	Session Key Selection . . . . .	47
4.5	Key Diversification . . . . .	50
6.1	Asset Management Based on RFID . . . . .	67
6.2	The EPCglobal Network . . . . .	69
7.1	RFID Middleware . . . . .	81
7.2	RFID Middleware Architecture . . . . .	83
7.3	Possible Entry Points . . . . .	85
7.4	Secure Architecture for a Distributed RFID Infrastructure . . . . .	96



# List of Tables

2.1	Security Services . . . . .	6
2.2	Security Threats and Countermeasures . . . . .	18
3.1	Barcodes vs. RFID [Fin03] . . . . .	22
3.2	Classification of RFID Systems [Fin03] . . . . .	23
3.3	Applications of RFID Standards . . . . .	31
3.4	Parts of the ISO/IEC 18000 Standard . . . . .	33
3.5	EPC Classes . . . . .	34
4.1	Attacks on RFID Systems and Countermeasures . . . . .	56
5.1	Threats to RFID Privacy and Countermeasures . . . . .	64
6.1	Security Mechanisms for MRTDs . . . . .	75
6.2	Comparison of Auto-ID Systems [Fin03] . . . . .	76

# List of Abbreviations

3DES	Triple DES
ACL	Access Control List
AES	Advanced Encryption Standard
API	Application Programming Interface
CRM	Customer Relation Management
DES	Digital Encryption Standard
DH	Diffie Hellmann
DMZ	Demilitarized Zone
DSP	Digital Signal Processor
EAN	Electronic Article Numbering
EAS	Electronic Article Surveillance
ECC	Elliptic Curve Cryptography
EPC	Electronic Product Code
ERP	Enterprise Resource Planning
FPGA	Field Programmable Gate Array
HMI	Human Machine Interface
IP	Internet Protocol
IPSec	Internet Protocol Security
ISO	International Organization for Standardization
JIT	Just in Time
JNI	Java Native Interface
KDC	Kerberos Domain Controller
MRTD	Machine Readable Travel Document
MRZ	Machine Readable Zone
NFC	Near Field Communication
NIDS	Network Intrusion Detection System
OCR	Optical Character Recognition
PJM	Phase Jitter Modulation
PKI	Public Key Infrastructure
RFID	Radio Frequency Identification
RSA	Rivest Shamir Adleman
SAM	Security Access Module
SAT	Sector Allocation Table
SSL	Secure Sockets Layer
TCP	Transport Control Protocol
TLS	Transport Layer Security
UbiComp	Ubiquitous Computing
WMS	Warehouse Management System
WSS	Web Service Security

# Chapter 1

## Introduction

In recent years automatic identification procedures (Auto-ID) have become very popular in many service industries, purchasing and distribution logistics, industry, manufacturing companies and material flow systems. Automatic identification procedures exist to provide information about people, animals, goods and products in transit. Barcode systems, which are predominantly used today, are based on the scanning and recognition of a barcode. While being cheap and easy to implement, their potential is limited by low storage capacities, line-of-sight requirements and the fact that they cannot be reprogrammed.

Radio Frequency Identification (RFID) makes use of radio frequency transmission to transfer data contactlessly between a reader and a tag. Even though RFID is more complex, it does not suffer from the drawbacks of optical identification and provides more functionality.

In a typical RFID system, tags storing a unique identifier and additional data are attached to objects or issued to people. When a tag or a group of tags is placed in the radio frequency field of a reader, the data contained in the tag's memory can be accessed by the reader. The data are usually preprocessed and passed on to enterprise applications by the RFID middleware. The RFID system, consisting of readers and tags, along with the RFID middleware and the enterprise applications is referred to as the RFID infrastructure. If readers are placed at several sites, the middleware has to be deployed across several servers and we speak of a distributed RFID infrastructure.

As the name implies the main functionality of RFID is the identification of objects. As a consequence the deployment of RFID can bring significant benefits wherever an automatic identification of items can reduce costs. This is particularly the case in the internal as well as in the global supply chain but also for asset management, access control systems and document verification systems.

The costs associated with the deployment of RFID can be split into fixed and variable costs. Fixed costs consist of server maintenance costs, software license costs as well as equipment depreciation and are largely constant and recurring. Tag costs are variable and increase with the number of tags issued. Since tags are usually issued in great numbers their costs are crucial in an investment analysis. Due to dropping tag prices and

standardization efforts, the use of RFID for mainstream applications is finally becoming profitable.

It is expected that RFID will be deployed on a large scale and by various companies and public services by 2010. At the same time modern IT applications show a trend towards increasing distribution, just-in-time networking and an associated rise in complexity. Due to distributed processing of data, transparency is lost, making frauds and abuse harder to detect. Since the benefits of RFID infrastructures come from an automated collection, processing and management of data, these systems have to satisfy tight security requirements. If an RFID system stores personal data, the privacy of users must also be protected. Additionally RFID systems allow the tracking of people and items based on the unique identifier on tags.

Security and privacy mechanisms for RFID systems usually affect tag, reader and middleware costs in various degrees. Due to the special cost characteristics of RFID systems, the employed security mechanisms have to offer a perfect trade-off between costs and efficiency.

The chapter "Security Primer" gives an introduction to security and well-known mechanisms for providing security in general. This comprises a presentation of the security services as described in the literature. Every secure system has to provide authentication, confidentiality, integrity and non-repudiation. Subsequently the basics of symmetric and asymmetric cryptography are described and the most important algorithms are presented and compared. A short introduction into digital signatures and Public Key Infrastructures (PKIs) is given. A close look at the recently established Web Services Security (WSS) standard as well as other security protocols such as Internet Protocol Security (IPSec) and Transport Layer Security (TLS) is also taken. Finally security threats and attacks on distributed systems are analyzed and evaluated.

The chapter "Radio Frequency Identification" introduces the reader to the fundamentals of RFID which are vital to understand in order to get an idea of the security threats that exist in RFID systems. First a link to barcode systems is established in order to understand the main purposes of RFID and current developments. Then RFID systems are classified based on their frequency range. In the next step a closer look on the components that make up an RFID system - readers, tags and infrastructure - is taken. There is also a section that deals with current standards and recent developments in the standardization process of RFID.

In the chapter "Security Aspects of RFID" RFID and the security mechanisms and requirements introduced in the security primer are put in relation. First the security services are revisited in an RFID context and possible fraud scenarios are identified. Then current RFID security mechanisms are presented and assessed in terms of cost, complexity and efficiency. At the end of this part the fraud scenarios in the context of RFID are weighed, ranked and evaluated in conjunction with the security mechanisms presented.

The chapter "RFID and Privacy" deals with privacy threats associated with a large scale deployment of RFID. The possible threats of RFID to privacy are presented and the most important privacy intrusion scenarios are identified. In the next step current

mechanisms and proposals for guaranteeing privacy are presented and assessed. With consideration of these mechanisms, the privacy threats are analyzed and evaluated.

The chapter "RFID Applications" presents current applications of RFID and evaluates their security on the air-interface as well as in their architecture. It is also shown how RFID can make business processes more efficient by supporting enterprise applications and adding services to the end-user. First the application areas of RFID are described and a close look at the benefits of RFID in these areas is taken. Subsequently particular implementations of RFID systems are presented and evaluated. Here the EPCglobal initiative is particularly important. Finally the chances and risks of RFID are summarized and an outlook on further developments is given.

The design and implementation of a secure RFID middleware architecture is dealt with in chapter "Securing a Distributed RFID Infrastructure". First the requirements on RFID infrastructures and most common architectures are explained. Then a threat model is created and security is introduced into the recommended architecture. This step consists of designing a secure network and choosing from the security services available. Another major issue dealt with is the key management which is highly complex and differs significantly from conventional distributed systems.

## Chapter 2

# Security Primer

Before security issues in RFID systems can be discussed it is necessary to get an understanding of the basic concepts and definitions in security. This chapter introduces the notion of the security services, the requirements for such services, as well as mechanisms for satisfying these requirements. Additionally, various symmetric and asymmetric encryption and digital signature algorithms are discussed and compared. Based on this introduction a closer look at common security protocols such as TLS/SSL, IPsec and Kerberos is taken. Also, the very recent Web Services Security (WSS) standard for guaranteeing message level security is discussed. Finally, this chapter analyzes typical attacks and classifies those attacks and threats.

### 2.1 Security Services

A security service is a collection of mechanisms, procedures and other controls that are implemented to help to reduce the risk associated with a specific threat to a system [BP01]. For example identification and authentication services help to reduce the risk posed by access to the system by an unauthorized user or attacker. Logging or monitoring is a service that helps to detect security breaches. For a specific application some security services might be more important and some less. The most important security services are:

1. Confidentiality, which ensures that data, software and messages are not disclosed to unauthorized parties.
2. Integrity, which ensures that unauthorized parties do not modify data, software and messages.
3. Authentication, which ensures that a network can only be accessed by individuals that are authorized.
4. Nonrepudiation, which ensures that entities involved in a communication cannot deny having participated in it.

5. Availability, which ensures that a service is available at all times.
6. Access Control, which ensures that network resources are being used in an authorized manner.

### **2.1.1 Authentication**

The first step in securing system resources is implementing a service to identify users by assigning them a unique user ID. Many other security services are based on this mechanism. For example logging services provide use information and access control permits access to a network based on user IDs. Obviously, unless the user is authenticated the system cannot trust the validity of the user's claim of identity. The user is authenticated by: an entity the user possesses (token), an entity only the user knows (password) or an entity that is unique to the user (fingerprint). Password-only systems are vulnerable to weak passwords chosen by users or the writing down of passwords. More security is provided by smart card based systems where possession of a token (smart card) and maybe the knowledge of a pin is required. Authentication is done with the challenge response scheme using real time parameters to prevent replay attacks and with encryption to avoid monitoring and capturing.

### **2.1.2 Confidentiality**

Sometimes facilitating access control by authentication is not possible because data may be sent via insecure channels. In this case the secrecy of information can be provided by confidentiality services. The use of encryption through symmetric and asymmetric ciphers or a combination of these two ciphers can reduce the risk of unauthorized disclosure of data by making it unreadable to unauthorized parties. Only the authorized user, who has the correct key can decrypt and read the data.

### **2.1.3 Integrity**

Data integrity services provide protection against intentional and accidental unauthorized modification of data. The services can be provided by cryptographic checksums or by highly granular access control and privilege mechanisms. Furthermore data integrity services help to ensure that a message is not altered, deleted or added during transmission. Most available security techniques cannot prevent the modification of a message, but they can detect that a message has been modified unless the message is deleted altogether.

### **2.1.4 Nonrepudiation**

Nonrepudiation helps to ensure that entities in a communication cannot deny having participated in all or part of the communication. Specifically the sending entity cannot deny having sent a message (nonrepudiation with proof of origin), and the receiving entity

cannot deny having received a message (nonrepudiation with proof of delivery). Nonrepudiation can be provided through the use of public key cryptographic techniques using digital signatures.

### 2.1.5 Summary

All of these services may be used to protect data in transit over a channel or at rest on a data storage. Table 2.1 summarizes the security services discussed.

Security Service	Attack	Solution
Authentication	Fake of identity	Passwords, tokens or a unique property
Confidentiality	Eavesdropping	Symmetric encryption, asymmetric encryption or both
Integrity	Modification of data	Checksums or Modification Detection Codes
Nonrepudiation	Fake of signatures	Public cryptographic techniques using digital signatures
Availability	Denial of Service attack	Redundancy, Quality of Service
Access Control	Unauthorized access	Hierarchical granular access and privilege architecture

Table 2.1: Security Services

## 2.2 Symmetric Cryptography

Symmetric cryptography, also known as secret key cryptography is based on encryption and decryption with the same key. The key and the plaintext are fed to an algorithm which generates the ciphertext. It is always assumed that the algorithm is known to the attacker but not the key. To automatically generate a key either a random number generator (RNG) can be chosen or a pseudo random number generator (PNRG). Often the latter is chosen because it is easier to implement. In that case, the PNRG must be seeded properly. Otherwise attackers could deduce the key. To avoid brute force attacks, where the attacker tries every possible key, a large enough bit size has to be chosen for the key. The algorithm can be broken by known plaintext attacks if the plaintext has some traceable relations to the ciphertext such as similar patterns. Usually algorithms create a key table based on a PRNG during key setup. This is done to support different key sizes and to prevent weak keys that can uncover algorithmic weaknesses.



### 2.2.1 Block Ciphers

Block ciphers break the plaintext into blocks usually 8 or 16 byte long and operate on them independently. Usually the last block is padded with the number of pad bytes added so that the receiver knows which bytes to discard. Multiple appearances of similar text also results in similar patterns in the ciphertext. This can be avoided by using feedback modes. The most common feedback mode is the cipher block chaining (CBC) mode where the current block of plaintext is XORed with the previous ciphertext. This adds no security but only ensures that similar blocks don't encrypt to the same ciphertext. Block cipher keys can be reused which is a great advantage if large amount of data as it is the case with databases needs to be encrypted. Furthermore it makes key management much easier. In contrast to stream ciphers block ciphers are more widely standardized and are a better choice if interoperability is required.

### 2.2.2 Stream Ciphers

Stream ciphers generate a pseudo random key stream based on the key and XOR it with the plaintext to generate the ciphertext. The key stream is independent from the input data. Decrypting is the same as encrypting because of the XOR function applied twice produces the original input. Stream ciphers are generally faster and use less code than block ciphers. The most common stream cipher RC4 is probably twice as fast as the fastest block cipher. Stream cipher keys should be used only once.

### 2.2.3 Symmetric Algorithms

Triple DES (3DES) is an adaption of the obsolete DES algorithm to meet modern security standards. It applies the DES algorithm 3 times and thus uses key lengths of 168 bits instead of 56 bits. Disadvantages of the 3DES algorithm are that encryption and decryption are very slow and that cryptanalysis has uncovered an algorithmic weakness reducing the effective key length to 108 bits. Many commercial replacements exist but none of these became a world wide standard comparable to DES. This led to the definition of the advanced encryption standard (AES), which includes the Rijndael algorithm. The Rijndael algorithm was invented by two Belgian researchers: Vincent Rijmen and Joan Daemen. Vincent Rijmen is professor at the Institute of Applied Information Processing and Communication at the Technical University of Graz. The AES is expected to become a worldwide standard within short time.

### 2.2.4 Password Based Encryption

Password based encryption (PBE) is a best practice when using symmetric cryptography. Data is encrypted with the session key, which is a number obtained from a PRNG seeded with several current parameters. After encryption the session key is encrypted with the key encryption key (KEK). The KEK is deduced from a password and then mixed with the salt, a random value based on several current parameters to prevent dictionary attacks.

The salt is then stored along with the encrypted session key. The major advantage of this approach is that the KEK doesn't have to be stored. If encrypted data needs to be shared the session key can be distributed and everybody can encrypt it with his own password. Another reason is that possible attackers need more data namely the encrypted data and the encrypted key. Only when the attacker has the encrypted session key he can launch an attack on the password which is easier than a brute force attack on the session key. The attacker also has to figure out in which way the password was mixed with the salt. Only then a dictionary attack can be launched. This step can be lengthened by mixing salt and password several times.

## 2.3 Asymmetric Cryptography

Asymmetric cryptography is also known as public key cryptography and applies two different keys. One key called the public key is used to encrypt data. The ciphertext can only be decrypted by the second key: the private key. A common practice is encrypting the plaintext with a symmetric algorithm and the session key with the public key of the recipient and send both of it. The recipient can then decrypt the session key with his private key and use it to decrypt the data. This is more efficient than encrypting all the data with the public key because asymmetric algorithms are considerably slower compared to symmetric algorithms and increase the data size .

### 2.3.1 Asymmetric Algorithms

The three most commonly used asymmetric algorithms are Rivest Shamir Adleman (RSA), Diffie Hellmann (DH), and Elliptic Curve Diffie Hellmann (ECDH).

The hard problem of RSA is factoring a number  $n$  as the product of two prime numbers. The concept of the Diffie-Hellmann algorithm is different in that it doesn't provide a means of encryption but a way to generate a symmetric session key based on the exchange of a public key. A DH public key is a prime number with size of the key length. After exchange the other party can use it to generate a temporary key pair which involves the use of a random number. The hard problem is discrete log problem which is mathematically related to the factoring problem. If one can be solved then the other too. So the same key lengths are used (typically 1024). A common size for the private value  $x$  is 160 bits in order to keep the number of calculations small.

ECDH is an algorithm that is based on scalar multiplication over an elliptic curve. The underlying problem of is known as the elliptic curve discrete log problem which is harder than the discrete log problem. Thus common key sizes are 160 to 170 bits. Key exchange is similar to Diffie-Hellmann except that both parties have to decide on which bits to take as a key because a point consists of an  $x$  and an  $y$  coordinate. Usually only the  $x$  coordinate is used. Elliptic curves can be used for encryption too but is rarely done due to performance hits.

With no further security measures the ECDH and the DH are susceptible to the man-in-the middle attack because an attacker could simply establish a connection with both parties. With RSA on the other hand it is not possible to decrypt any data without the private keys. ECDH and DH require both users have to choose a (good) key in RSA only one. ECDH provides a greater per bit security than RSA but is generally slower. A 1024 bit RSA key matches a 160 bit ECDH key. The RSA algorithm is about 18 times faster in public key operations (initiating contact) and about twice as slow as ECC in private key operations (receiving). ECC can be accelerated with lookup tables requiring storage space of 20 kB. Transmission size is the same as the key size for RSA and DH. For ECC it is twice the size. That means with each a digital envelope is sent, RSA adds 1024 bits and ECC 320.

## 2.4 Digital Signatures

Digital signatures are used to authenticate the author of a message and to prevent people from going back on their electronic word (nonrepudiation). Many nations are passing laws that make digital signatures legally accepted. Digital signatures are based on the fact that data encrypted with a private key can only be decrypted with the public key belonging to it.

Because public key cryptography is slow usually not the whole message is signed but only a hash of the message also called message digest. Hashes are regardless of the size of the plaintext always the same size and meet some important requirements:

1. Pre-image resistance: it should be computationally impossible to find a pre-image to a given hash value.
2. 2nd pre-image resistance: it should be computationally impossible to find a second pre-image to a given input.
3. Collision resistance: it should be computationally impossible to find two different inputs with the same hash value.

Although collisions are theoretically possible for good algorithms no two similar hashes have ever been found. The basic principle with hashes is that a message where a hash value has been appended cannot be altered without recalculating and usually signing the hash. Important hash algorithms are MD5 producing 128 bit digests. The SHA-1 algorithm looks very similar to MD5 but is more secure and produces 160 bit hashes. Message digests also guarantee that data has not been altered.

The sender of a message computes a hash and signs it by encrypting it with his private key. At the end of the line the receiver separates the signature from the message and computes the hash using the same algorithm. Then he decrypts the signature with the sender's public key and compares the hash values to verify the signature and the integrity of the data.

### 2.4.1 Digital Signature Algorithms

The most commonly used digital signing algorithms are RSA, DSA and ECDSA, which is the same as DSA but based on elliptic curves. The process of signing with RSA is encrypting the hash with the private key. Third parties can then verify the signature by decrypting with the public key and comparing the hash. To forge a signature the private key has to be found. With DSA the signer digests the message with SHA-1 and uses the digest as a number. This number along with a pseudo-random number  $k$  and the private key is fed to the DSA algorithm, which produces a pair of numbers:  $r$  and  $s$ . On the other side the verifier computes the SHA-1 digest of the message. He feeds the digest, the number  $s$  and the public key to the algorithm and compares the result  $v$  to  $r$ . If  $v$  is the same as  $r$  the result is verified. This system is very similar to the Diffie-Hellmann algorithm. The ECDSA algorithm looks a lot like DSA except that it is based on elliptic curves. The signer has three inputs: the digest,  $k$  and the private key. The output is  $v$ . If  $v$  and  $r$  are the same the signature is verified otherwise something is wrong.

The fastest algorithm for verifying signatures is RSA. Signing is two times faster with ECC and about six times faster with ECC with acceleration. DSA and ECDSA signatures are 340 bits regardless of the key size. An RSA signature's length is the same as the key size's, usually 1024 bits.

## 2.5 Public Key Infrastructures

The distribution of public keys is often problematic because attackers could replace someone else's public key with their own. A solution to this is the use of public key infrastructures. A public key certificate (PKC) binds a user to his public key. Certification authorities (CA), issue certificates by taking a user's name, his public key and other identifying information and sign it with their private key. The verifier takes the PKC from a certificate directory, which is usually associated with the CA, and verifies it with the CA's public key.

Before issuing a certificate trust has to be established between the CA and the user. After that step has been carried out there are two ways to initiate a certificate request. Users can have the CA generate a key pair for them or can do it themselves and hand in the public key in the form of a PKCS #10 certificate signing request. Certificates are issued to be valid and usable until the date indicated in the validity field. In case the certificate needs to be stopped from use it needs to be revoked. This is usually done by the use of certificate revocation lists (CRL). The CLR contains all certificates that have been revoked or are currently on hold.

## 2.6 Security Protocols

Now that an overview of security services and security mechanisms has been given, this section will now take a closer look at several implementations of those mechanisms and

how they provide the security services.

### 2.6.1 Internet Protocol Security (IPSec)

The Internet Protocol Security is a framework of open standards for securing communications over IP networks. It provides security at the network layer (see figure 2.1). Consequently there is no need to adapt applications to IPSec. IPSec encrypted packets look like ordinary IP packets and can easily be routed through any IP network.

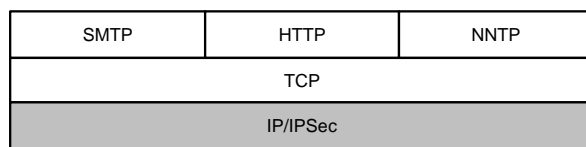


Figure 2.1: Network Security

The only devices that know about the encryption are the endpoints. A number of security protocols are supported providing the following services:

- Access control,
- connectionless integrity,
- data origin and authentication,
- rejection of replayed attacks,
- confidentiality (encryption) and
- limited traffic-flow confidentiality.

These services are provided by two protocols: the authentication header (AH) protocol and the encapsulating security payload (ESP) protocol. The AH protocol supports access control, data origin authentication, connectionless integrity and the rejection of replay attacks. AH uses a keyed hash function because digital signature technology is too slow. However with the AH no confidentiality protection is provided. This is in the responsibility of the encapsulating security payload (ESP) protocol which provides confidentiality services for IP data while in transit. Optionally it can also provide authentication services. The key used for encryption is associated with the SPI.

Before communicating, a security association (SA) has to be established by agreeing on the parameters to be used. This is taken care of by the Internet Key Exchange (IKE). There are three operating modes for IKE. In Main Mode both parties agree on parameters to be able to communicate securely long enough to set up an SA for future communication.

Three two way messages are exchanged. In the first exchange initiator and responder agree on basic algorithms and hashes. In the second public keys are exchanged for a Diffie-Hellmann exchange and random numbers are passed to be signed and verified in the third step. Aggressive mode can be used to reduce the number of exchanges to two. Although aggressive mode is faster, it does not provide identity protection. After a secure connection already exists the much faster Quick Mode can be used to agree on a new password.

The main advantages of IPsec are that it is

1. usable for IPv4 as well as IPv6,
2. transparent to applications,
3. independent from other security mechanisms, and that it
4. defines no rigid security architecture and
5. allows for the definition of a variable security policy.

### **2.6.2 Secure Sockets Layer (SSL)**

SSL provides session based encryption and authentication, establishing a secure pipe between two parties: the client and the server. In an SSL communication the server and optionally the client are authenticated to avoid eavesdropping, tampering and message forgery in client-server applications. SSL also supports privacy by establishing a shared secret between the two parties. In contrast to IPsec, SSL works at the transport layer (see figure 2.2) and is independent of the application protocol used. SSL provides the following features:

1. Authentication of the server against client.
2. Authentication of the client to the server (optional).
3. Encrypted connection between client and server.
4. Public key encryption is used to generate a session key.

Each endpoint in an SSL connection must implement the matching side of the protocol. The interaction between the state machines at the server and the client is called the Handshake. The SSL handshake protocol is responsible for coordinating the states of the client and the server. An SSL session can include multiple secure connections and parties can have multiple simultaneous sessions.

Data transmitted and received from upper application layers is processed at the SSL record layer. At first the data is fragmented into smaller blocks with a maximum size of 16K. Optionally the blocks are compressed with the algorithm defined by the current session state. Then a MAC is computed using the previously established shared secret key.

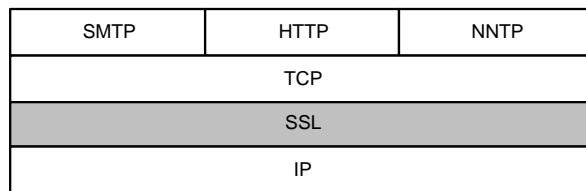


Figure 2.2: Transport Security

The block and the MAC are encrypted with the symmetric cipher defined by the session state. Finally a header is added including the content type, version data, and the block length. For more information on SSL and TLS see [FKK96] and [DA99]. Disadvantages of SSL are that there is no support for UDP and that the use of SSL leads to performance penalties.

### 2.6.3 Kerberos

The Kerberos protocol is based on the key distribution model developed by Needham and Schroeder [NS78], and defines how clients interact with a network authentication service. Clients obtain tickets from the Kerberos Key Distribution Center (KDC), and they present these tickets to servers when connections are established. The tickets represent the client's network credentials. Kerberos provides the following security services:

1. Mutual authentication,
2. delegated access control,
3. privacy and
4. data integrity.

The protocol consists of the following steps, which are illustrated in figure 2.3:

**Authentication exchange:** The client asks the authentication server for a ticket to the ticket-granting server (TGS). The authentication server looks up the client in its database, then generates a session key (SK1) for use between the client and the TGS. Kerberos encrypts the SK1 using the clients secret key. The authentication server also uses the TGSs secret key, which is known only to the authentication server and the TGS, to create and send the user a ticket-granting ticket (TGT).

**Ticket-granting service exchange:** The client decrypts the message and recovers the session key, then uses it to create an authenticator *Auth* containing the users name, IP address and a time stamp. The client sends this authenticator, along with the

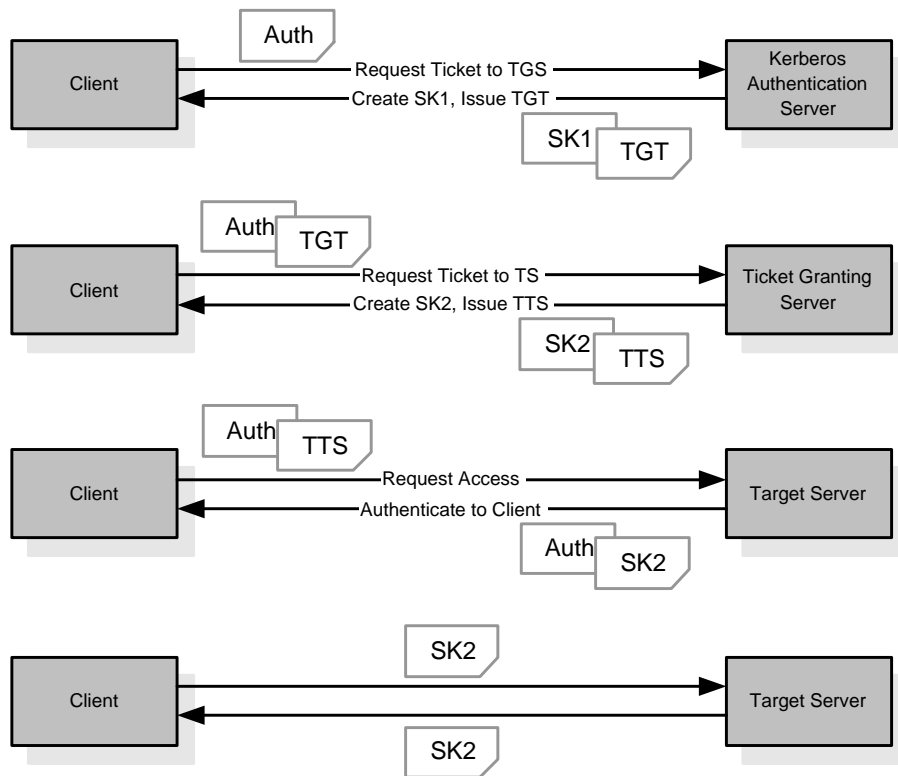


Figure 2.3: The Kerberos Protocol

TGT, to the TGS, requesting access to the target server. The TGS decrypts the TGT, then uses the SK1 inside the TGT to decrypt the authenticator. It verifies information in the authenticator, the ticket, the clients network address and the time stamp. If everything matches, it lets the request proceed. Then the TGS creates a new session key (SK2) for the client and target server to use, encrypts it using SK1 and sends it to the client. The TGS also sends a new ticket TTS containing the clients name, network address, a time stamp and an expiration time for the ticket all encrypted with the target servers secret key and the name of the server.

**Client/server exchange:** The client decrypts the message and gets SK2. The client creates a new authenticator encrypted with SK2 and sends the session ticket (already encrypted with the target servers secret key) and the encrypted authenticator. Because the authenticator contains plaintext encrypted with SK2, it proves that the client knows the key. The encrypted time stamp prevents an eavesdropper from recording both the ticket and authenticator and replaying them later. The target server decrypts and checks the ticket, authenticator, client address and time stamp. For applications that require two-way authentication, the target server returns a



message consisting of the time stamp plus 1, encrypted with SK2. This proves to the client that the server actually knew its own secret key and thus could decrypt the ticket and the authenticator.

**Secure communications:** The target server knows that the client is who he claims to be, and the two now share an encryption key for secure communications. Because only the client and target server share this key, they can assume that a recent message encrypted in that key originated with the other party.

Currently Kerberos is the only widely-adopted authentication protocol that is capable of performing delegation. When n-tier applications are not using an authentication protocol that enables delegation, usually user identifiers are passed through application message bodies. Without further security measures to protect the application message while it is in transit, this approach is susceptible to message modification. In addition to delegation, Kerberos also offers the benefits of mutually-authenticating communicating parties, as well as protecting the application message when it is in transit by using data encryption.

#### 2.6.4 Web Services Security (WSS)

The Web Services Security (WS-Security) [OAS04] specification defines a set of SOAP header extensions for end-to-end SOAP messaging security. It supports message integrity and confidentiality by allowing communicating partners to exchange signed and encrypted messages in a Web services environment. WS-Security is flexible and is designed to be used as the basis for the construction of a wide variety of security models including PKI, Kerberos, and SSL. Specifically WS-Security provides support for multiple security tokens, multiple trust domains, multiple signature formats, and multiple encryption technologies.

The Web services security language must support a wide variety of security models. The following list identifies the key requirements for this specification:

- Multiple security tokens for authentication or authorization
- multiple trust domains,
- multiple encryption technologies and
- end-to-end message-level security and not just transport-level security.

The following topics are not part of the specification:

- Establishing a security context or authentication mechanisms that require multiple exchanges,
- key exchange and derived keys and
- how trust is established or determined.

Protecting the message content from being intercepted or illegally modified are primary security concerns. This specification provides a means to protect a message by encrypting and/or digitally signing a body, a header, an attachment, or any combination of them.

Message integrity is provided by leveraging XML Signature in conjunction with security tokens to ensure that messages are transmitted without modifications. The integrity mechanisms are designed to support multiple signatures, potentially by multiple actors, and to be extensible to support additional signature formats.

Message confidentiality leverages XML Encryption in conjunction with security tokens to keep portions of a SOAP message confidential. The encryption mechanisms are designed to support additional encryption processes and operations by multiple actors.

## 2.7 Security Threats in Distributed Systems

There are various groups of attackers following different goals. These range from script kiddies and the disappointed employees to cyber-terrorists and criminals. However, sophisticated attacks have a common underlying pattern and can be classified by their goals.

### 2.7.1 Anatomy of Attacks

Most of the more sophisticated attacks follow a general pattern (see figure 2.4), which starts with the *surveying and assessing* of the target's characteristics. These characteristics may include supported services and protocols as well as potential vulnerabilities and entry points. Based on the information gained in this step an attack can be planned.

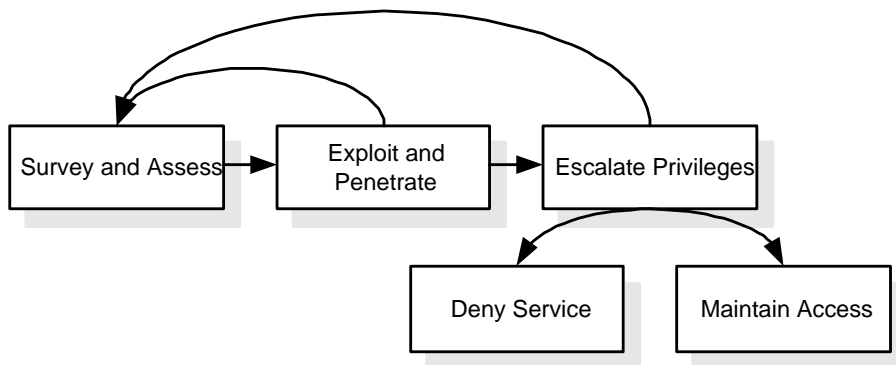


Figure 2.4: Anatomy of Network Attacks

After the target has been thoroughly scanned, the next step is the *exploiting and penetrating* of the network. If network and hosts are properly secured the application becomes the next target of the attack. For an attacker the easiest way into an application

is through the same entrance legitimate users use, for example through the logon page or a page that doesn't require authentication.

When a network or an application has been compromised, for example by injecting code or creating a session, the attacker immediately attempts an elevation of privileges. Particularly administration privileges or high level privileged system accounts are looked for. A primary defense against privilege escalation attacks is the use of least privileged accounts.

Once access has been gained to a system, an attacker usually attempts to make future access easier and to cover traces. Common approaches for *maintaining access* include the planting of back-door programs or using an existing account that lacks strong protection. Covering tracks includes the clearing of logs and the use of hiding tools. Therefore log files should be secured and analyzed on a regular basis as they can uncover early signs of an attack and prevent further damage.

Attackers who cannot gain access to a system may mount a *Denial of Service* attack to prevent others from using an application. Sometimes denial of service is the primary goal of an attack. A typical example for an attack of this kind is the SYN flood attack which jams the TCP stack and prevents others from access.

## 2.7.2 Classification of Attacks

While there are many variations of specific attacks and attack techniques, their goals can be reduced to the following list:

1. *Spoofing* is the use of a false identity for example to get access to a system. This can be done by using stolen user credentials or false IP addresses.
2. *Tampering* is the unauthorized modification of data, for example as it flows over a network between two computers.
3. *Repudiation* is the ability of users to deny that they performed specific actions or transactions. These attacks are difficult to prove.
4. *Information disclosure* is the unwanted exposure of private data. For example, a user views the contents of a table or file he or she is not authorized to open, or monitors data passed in plaintext over a network.
5. *Denial of service* is the process of making a system or application unavailable.
6. *Elevation of privilege* occurs when a user with limited privileges assumes the identity of a privileged user to gain privileged access to an application.

Each threat category described in the list has a corresponding set of countermeasure techniques that are presented in table 2.2 and should be used to reduce risk.

Threat	Countermeasure
Spoofting	Strong authentication Store of secrets in encrypted or hashed form No passing of credentials in plaintext over network
Tampering with data	Data hashing and signing Digital signatures Use of strong authorization Use of tamper-resistant protocols over the network Secure communication links providing message integrity
Repudiation	Secure audit trails Digital signatures
Information Disclosure	Strong authorization Strong encryption Use of protocols that provide confidentiality
Denial of Service	Resource and bandwidth throttling Validation and filtering of input
Elevation of Privilege	Principle of least privilege

Table 2.2: Security Threats and Countermeasures

### 2.7.3 Network Threats

The primary components that make up a network infrastructure are routers, firewalls, and switches. They act as the gatekeepers guarding servers and applications from attacks and intrusions. An attacker may exploit poorly configured network devices. Common vulnerabilities include weak default installation settings, wide open access controls, and devices lacking the latest security patches.

#### Information Gathering

Network devices can be discovered and profiled in much the same way as other types of systems. Attackers usually start with port scanning. After open ports are identified techniques such as banner grabbing and enumeration can be used to detect device types and operating system and application versions. This information may reveal known vulnerabilities that may not be updated with security patches.

As a countermeasure routers should be configured to restrict their responses to footprinting requests. Operating systems that host network software should be configured to prevent footprinting by disabling unused protocols and unnecessary ports.

## **Sniffing**

Sniffing or eavesdropping is the act of monitoring traffic on the network for data such as plaintext passwords or configuration information. With a simple packet sniffer an attacker can easily read all plaintext traffic. Also attackers can crack packets encrypted by lightweight hashing algorithms and can decipher the payload considered to be safe. The sniffing of packets requires a packet sniffer in the path of the server/client communication.

To prevent sniffing strong physical security and proper segmenting of the network is vital. This is the first step in preventing traffic from being collected locally. Communication should be encrypted fully, including authentication credentials. This prevents sniffed packets from being usable by an attacker. SSL or IPsec can be used.

## **Spoofing**

Spoofing is a means to hide one's true identity on the network. To create a spoofed identity, an attacker uses a fake source address that does not represent the actual address of the packet. Spoofing may be used to hide the original source of an attack or to work around network access control lists (ACLs) that are in place to limit host access based on source address rules.

Although carefully crafted spoofed packets may never be tracked to their original sender, a combination of filtering rules prevents spoofed packets from originating from the local network, allowing the blocking of obviously spoofed packets. Countermeasures to prevent spoofing include the filtering of incoming packets that appear to come from an internal IP address and the filtering of outgoing packets that appear to originate from an invalid local IP address.

## **Middleman Attacks**

A middleman attack occurs when someone between two communicating parties is actively monitoring, capturing, and controlling the communication without the knowledge of the users. For example, an attacker can negotiate encryption keys with both users. Each user then sends encrypted data to the attacker, who can decrypt the data. When computers are communicating at low levels of the network layer, the computers might not be able to determine with which computers they are exchanging data.

Middleman attacks can be prevented by the use of encrypted session negotiation and encrypted communication channels. Information of platform patches to fix TCP/IP vulnerabilities, such as predictable packet sequences may also be important.

## **Denial of Service**

Denial of service denies legitimate users access to a server or services. The SYN flood attack is a common example of a network level denial of service attack. It is easy to launch and difficult to track. The aim of the attack is to send more requests to a server

than it can handle. The attack exploits a potential vulnerability in the TCP/IP connection establishment mechanism and floods the server's pending connection queue.

Countermeasures to prevent denial of service include the increasing of the size of the TCP connection queue and the decreasing of the connection establishment period. Employing dynamic backlog mechanisms ensures that the connection queue is never exhausted. A Network Intrusion Detection System (NIDS) can automatically detect and respond to SYN attacks.

## 2.8 Summary

Security services are a collection of mechanisms, procedures and other controls that are implemented to help reduce risks of a specific threat. The central security services are: confidentiality, integrity, authentication, nonrepudiation, availability and access control.

There are various mechanisms to ensure that the security services can be guaranteed. Symmetric cryptography is based on encryption and decryption with a single key. The fact that there is always the danger of a key compromise when exchanging keys is known as the key exchange problem. Asymmetric algorithms solve this problem by providing a private key as well as a public key. The problem with asymmetric cryptography is that algorithms are usually more complex and more costlier to implement than their symmetric counterparts. Cryptography can provide confidentiality and integrity. Digital signatures are another mechanism to provide integrity and nonrepudiation, which is based on hashes and keys. First a hash of the message is calculated which is then signed with a private key. Verifiers can then recompute the hash and verify the correctness of the signature. Either symmetric or asymmetric keys can be used. In all cases where asymmetric keys are used there needs to be a mechanism to ensure the correct distribution of public keys. This is achieved with certificates in public key infrastructures. Several protocols exist that make use of symmetric and asymmetric algorithms to provide all or a subset of the security services described. These include protocols providing transport level security such as SSL and IPSec, protocols providing message level security such as WSS and security services provider such as Kerberos.

There are many variations of specific attacks and attack techniques. The goals of this attack can be: spoofing, tampering, repudiation, information access, denial of service or the elevation of privilege.

## Chapter 3

# Radio Frequency Identification

The goal of this chapter is to make the reader familiar with the basics of RFID. It starts by highlighting the current evolution of automatic identification from barcodes to RFID and compares the existing auto-id systems. Subsequently current RFID systems are classified and compared. After a system overview has been given, the technical background of RFID readers and tags is presented. Finally, properties of the various RFID standards are discussed at the end of this chapter.

### 3.1 From Barcodes To RFID

Barcodes are predominantly used for identifying and tracking products throughout the supply chain. Even though they can achieve efficiencies in the order of 90% [Fin03], they still show some limits in the technology, for which RFID is able to provide a better solution and further optimization, as it is shown in table 3.1.

Bar coding is a cost-effective and low-risk method of encoding information. RFID on the other hand enables users to collect and encode information for many items simultaneously with no line-of-sight requirement. Unlike bar codes, for which many standards already exist, RFID is just at the beginning of standardization. There are common frequency ranges for example, but the reader power output and specific frequency may vary by country and manufacturer. In addition, systems within the frequency range may have their own chip set, protocol for memory storage, air protocol and antenna design. With no-contact, no-line-of-sight reading, the RFID tag's position isn't as crucial as it is for barcodes. Furthermore RFID tags are more robust than barcodes in foggy and dusty environments. With decreasing equipment and tag costs, RFID gains competitive edge over barcodes. However most analysts today predict that barcodes and RFID systems will coexist for a long time.

System Parameters	Barcode	RFID
Data quantity (bytes)	1..100	16..64k
Data density	Low	Very high
Machine readability	Medium	Good
Rewritable	Yes	No
Influence of dirty/damp	Very high	No influence
Influence of covering	Total failure	Moderate
Data carrier cost	Very low	Medium
Reading electronic cost	Low	Medium
Security features	None	Available
Multiple reading, anticollision system	No	Yes
Reading speed	Low	Very fast
Max. distance between data carrier and reader	0-50cm	0-5m

Table 3.1: Barcodes vs. RFID [Fin03]

## 3.2 Classification of RFID Systems

RFID systems exist in many different variations. In order to provide an overview of RFID systems, we can look on special features, that differentiate those systems. Possible criteria include operating mode, frequency range and coupling or information processing capabilities. Furthermore differentiation based on modulation technique, data quantity and power supply is possible.

### 3.2.1 Frequency, Range and Coupling

RFID systems are usually classified by three parameters: operating frequency, range and coupling. Operating frequencies for RFID systems range from 135kHz (low frequency band) to 5,8 GHz (microwave). Physical coupling can either be electrical, magnetic or electromagnetic. Ranges from a few mm to 15 m and more can be achieved. Table 3.2 gives a classification of RFID systems based on their frequencies.

RFID systems with very small ranges of up to 1cm are called close coupling (CC) systems. Here the transponder has to be inserted into a reader or held very close to the surface of a reader. Close coupling systems use either magnetic or electric fields and can operate on any frequencies from DC to 30Mhz. This allows providing the transponder with a great amount of energy. Close coupling systems are used where there is a great demand for security and low ranges can be used such as door locks or cash cards. Currently close coupling contactless chip cards are only offered in the ID1 format (ISO 10536) and play a declining role on the market.

When read and write ranges go up to 1m the systems are called remote coupling



	LF	HF	UHF	Microwave
Frequency	125-134kHz	13,56MHz	868MHz 915MHz	2,45MHz 5,8GHz
Reading Range	up to 1m	up to 1m	up to 4m	up to 15m
Reading Speed	slow	medium	fast	very fast
Influence of humidity	none	none	negative	negative
Influence of metal	negative	negative	none	none
Alignment of Transponder for reading	not necessary	not necessary	sometimes necessary	always necessary
Internationally accepted frequency	yes	yes	EU,USA	only USA
ISO Standards	11784/85 14223	14443 15693 18000	15693 18000	18000

Table 3.2: Classification of RFID Systems [Fin03]

(RC) systems. These systems primarily use inductive or magnetic coupling, sometimes capacitive or electric coupling is used. There exist a great number of standards for RC systems including the ISO 14443 proximity coupling, and the ISO 15693 vicinity coupling standards. Operating frequencies are below 135 kHz or 13,56 MHz. The 13,56 MHz band is particularly interesting because there exist no regulations for it and no license costs have to be paid.

RFID systems with ranges exceeding 1m are called long range (LR) systems. All LR systems operate with electromagnetic waves in the UHF and microwave band. The majority of these systems are backscatter systems. In addition there exist some surface wave transponders in the microwave range. All of these systems are operated on the frequencies 868 MHz (Europe) and 915 MHz (USA) and in the microwave area at 2,5 GHz, and 5,8 GHz. With passive, battery-less backscatter systems ranges of typically 3m can be achieved. With active, battery-powered transponders ranges of up to 15m are possible. In active transponders the battery's sole purpose is the supplying of the chip with energy as well as data retention. Data transmission is done via the electromagnetic field radiated from the reader.

### 3.2.2 Information Processing in the Transponder

RFID systems can also be classified according to the range of information and data processing functions offered by the transponder and the size of its data memory. The two ends of this spectrum are represented by low-end and high-end systems.

## **Low-end Systems**

The bottom end of low-end systems is represented by EAS (Electronic Article Surveillance) systems. These systems only check for the possible presence of a transponder in the interrogation zone of a reader. Read-only transponders with a microchip are also classified as low-end systems. The data on these tags, usually a serial number, is permanently encoded. If a read-only transponder is placed in the HF field of a reader, the transponder begins to continuously broadcast its own serial number. There is a unidirectional flow of data from the transponder to the reader. It also has to be made sure that there is only one transponder in the readers interrogation zone because two or more transponders transmitting at the same time would lead to a data collision. Low-end systems are characterized by low power consumption and low manufacturing costs.

## **Mid-range Systems**

The mid-range is represented by systems with writable data memory. In this sector the greatest diversity of types can be found. Memory sizes range from a few bytes to over 100 kbyte. Transponders are able to process simple reader commands for selective reading and writing of the data memory in a permanently encoded state machine. In general, the transponders also support anti-collision procedures, so that several transponders located in the readers interrogation zone at the same time do not interfere with each another and can be selectively addressed by the reader. Cryptographic procedures, such as authentication between transponder and reader, and data stream encryption are also common in these systems. These systems are operated at all frequencies available to RFID systems.

## **High-end Systems**

The high-end segment comprises systems with a microprocessor and a smart card operating system. The use of microprocessors facilitates the realization of more complex encryption and authentication algorithms than would be possible using state machines. The top end of high-end systems is occupied by modern dual interface smart cards, which have a cryptographic coprocessor. The enormous reduction in computing times that results from the use of a coprocessor means that contactless smart cards can even be used in applications that set high requirements on the secure encryption of the data transmission, such as electronic purses or ticketing systems for public transport. High-end systems are almost exclusively operated at the 13.56 MHz frequency.

### **3.2.3 Other Classification Criteria**

RFID systems operate according to one of two basic procedures: full duplex (FDX), half duplex (HDX) systems and sequential systems (SEQ). In full and half duplex systems the transponders response is broadcast when the readers RF field is switched on. Because the transponders signal to the receiver antenna can be extremely weak in comparison with

the signal from the reader itself, appropriate transmission procedures must be employed to differentiate the transponders signal from that of the reader.

The amount of data that can be stored on a transponder is also a classification criteria of RFID systems. Data capacities start with one bit and go up to several hundred kbytes.

Power supply in transponders can either be active or passive. An active transponder has a battery that supplies it with power. Passive transponders on the other hand receive their power from the RF field emitted by the reader.

Another distinguishing feature is the memory type. The data storage can be realized in three different ways:

**EEPROM:** Inductive coupling systems are generally working with EEPROMs. Drawbacks are the high power consumption and the limited number of write processes.

**FRAM:** With this new technology power consumption less than a factor of 100 compared to EEPROM technology can be achieved. The write delay time is also about a factor of 1000 less.

**SRAM:** Particularly in microwave systems static RAM (static random access memory, SRAM) is used for data storage with very short write cycles. SRAMs have a permanent power consumption and can therefore only be used in active transponders.

There exist different technologies for transferring data from the transponders to the reader. In reflection or backscatter systems the tags modulate a reflected wave. Readers need a directional coupler to extract the tag's response. With load modulation the reader field is influenced by the transponder, which modulates its information with a resistor. Alternatively a sub-harmonic or a harmonic wave is created and modulated by the transponder.

### 3.3 RFID Transponders

Data storage in transponders can either be based on an integrated circuit or physical effects. Integrated circuit transponders are further classified into memory with state machine and programmable microprocessor. The 1 bit transponder and the surface wave transponders belong to the group of transponders based on physical effects.

#### 3.3.1 Transponders with Memory

Transponders with memory range from simple read-only transponders to high-end transponders with sophisticated crypto-functions. As a memory RAM, ROM, EEPROM or FRAM may be used. The RF interface is needed for power supply and reader communication. Security functions and addressing are carried out by the address and security logic.

The *RF interface* is the bridge between the analogue high frequency channel from the reader and the digital integrated circuit in the transponder. It carries out the function of a

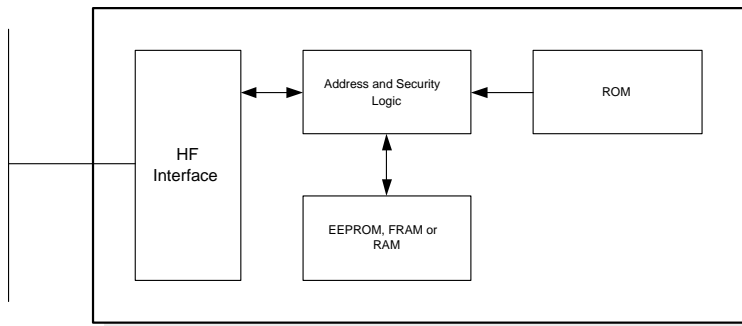


Figure 3.1: RFID Tag with Memory

modulator-demodulator (modem). System clock is obtained from the carrier frequency of the RF field. To send data back to the reader, a load modulator, a backscatter modulator or other techniques are used. Passive transponders draw energy from the RF field of the reader and supply it to the integrated circuit after rectification.

The *address and security logic* is the core of the transponder, which manages all procedures on the chip. The power-on logic guarantees that the transponder enters a defined state after entry into the HF field of the reader and sufficient power is supplied. Special I/O registers are used for communication with the reader. Optionally there can be a crypto-unit for authentication, ciphering and key management embedded on the chip. The memory consisting of ROM for fixed data storage and EEPROM or FRAM for variable data storage is connected to the address and security logic with the data bus and the address bus. The system clock is obtained from the RF field of the reader and supplied to the address and security logic. The state dependent control flow is managed by a state machine. Complexities comparable to microprocessor solutions can be achieved although control flow is fixed. An adaption for special situations can only be achieved by a design change. Therefore this solution is interesting for applications with a great number of transponders.

There are three types of different *memory architectures* with the read only transponder being the low-end and low-cost solution. If the read only transponder enters a reader field it begins to continuously transmit its unique ID which is assigned to the chip at production time. Communication only takes place in one direction. Read/write transponders can be filled with data from a reader. Commonly a fixed number of bytes are combined to a block. Only blocks of data can then be read and written. This allows easier addressing in the reader and the chip. If data need to be changed the corresponding blocks have to be read out. Then the bits are changed and the blocks are rewritten to the transponder.

Transponders with dual port EEPROMs can be operated via the RF field or the I2C bus. The I2C bus was first designed for the communication between microprocessors on the same circuit board and requires only two wires: SDA (serial data) and SCL (serial

clock). For access to the EEPROM two independent state machines are required. An arbitrage logic avoids conflicts at synchronous access from the RF interface and the I2C interface. It makes sure the other interface is locked for the time the other one carries out an operation. Here the access register can be used to set different access rights for the I2C interface and the RF interface.

### 3.3.2 Transponders with Microprocessor

In contactless microprocessors tags the state machine is replaced by a microprocessor. Optionally mathematic coprocessors for example for cryptographic functions can be embedded. Just as it is the case with contact cards, contactless chip cards use a operating system (OS) of there own. The OSs task is the management of data transmission from and to the chip card, sequence control, data management and execution of cryptographic functions. Commands from the reader are received via the RF interface. Error detection and correction is carried out by the I/O manager. The secure messaging manager decipheres the command and checks its integrity. Subsequently the command interpreter tries to decode the command. If that step fails the return code manager sends the corresponding return message via the I/O manager. Else the decoded command is executed. Access to the EEPROM is carried out exclusively by the file management and the data management. The file manager also checks the access conditions.

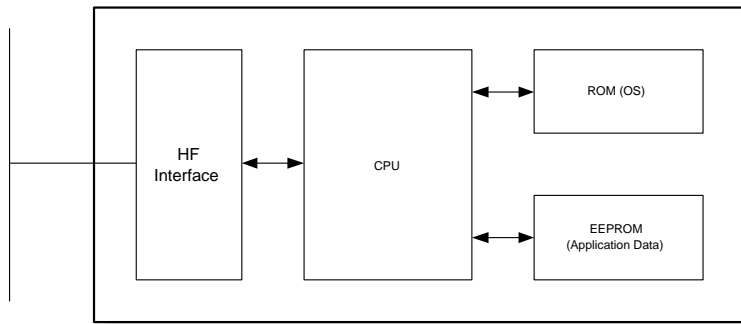


Figure 3.2: RFID Tag with Microprocessor

Dual interface cards are based on the trend to combine contact cards, which have so far been used for payment and high security applications with user-friendly and fast contactless card applications. From the view of the application it should not matter which interface is currently used. They should be interchangeable. ISO7816, the standard for secure messaging, makes sure that security is also the same for both interfaces by preventing replay and fraud. The main difference is the available power which must be taken care of by special low voltage designs and power management units. No explicit switching between contact and contactless operating mode is needed. Ticketing applications usu-

ally require transaction times of at most 100ms. To achieve this as well as the specified security, crypto-units are used as coprocessors. These coprocessors can speed up DES encryption about 100 times. The CPU just has to write data and keys to special registers and start calculations with the control register. In the near future asymmetric algorithms will become increasingly important. As a result some chips have asymmetric cryptography coprocessors included.

### 3.4 RFID Readers

There are different kinds of readers depending on the technical system used. However all readers can be reduced to two functional blocks: the HF-interface, consisting of transmitter and receiver, and the control unit.

The tasks of the *RF interface* are the emission of a RF field for energy supply for the transponder, modulation of the transmitter signal for the transmission of data to the transponder and the demodulation of data received from the transponder. There are two different branches for data sent to the transponder and for the data received, called transmitter branch and receiver branch. Full duplex (FDX) systems can send and receive at the same time while half duplex (HDX) systems can only do one at a time. Sequential or pulsed systems provide the RF field in pulses to supply the transponder with energy and to transmit data to the transponder. Breaks in energy supply are used to transmit data from the transponder to the reader.

Communication with the application as well as the execution of commands are taken care of by the *control unit*. The control unit also manages communications with the transponder according to the master/slave principle and does the encoding and decoding of the signal. In more complex systems the control unit also executes an anti-collision algorithm, deciphers and enciphers transmitted data and manages an authentication procedure between transponder and reader. To carry out these tasks efficiently, the control unit has a microprocessor in its core. Crypto-functions and signal coding are usually delegated to an additional module to relieve the microprocessor. Access to these modules is done via a register based microprocessor bus. For communication between the application and the reader in most cases an RS232 or an RS485 interface is used. High-end readers also support TCP/IP or USB communications. The interface between the control unit and the RF interface contains a binary representation of the state of the RF interface.

### 3.5 Dataflow in RFID Systems

For transponder access an application needs a reader as an interface. All communication is based on the master/slave principle meaning that all activities are initiated by the application software, which acts as the master. The reader acts as a slave. To execute a command from the application software, the reader establishes a connection with the tag. Then the reader acts as a master and the transponder as a slave. The usual communication process

consists of the activation of the transponder, authentication and data transmission. Consequently the main tasks of the reader are the activation of transponders, the establishing of a link and the transport of data between transponder and reader, authentication and anti-collision.

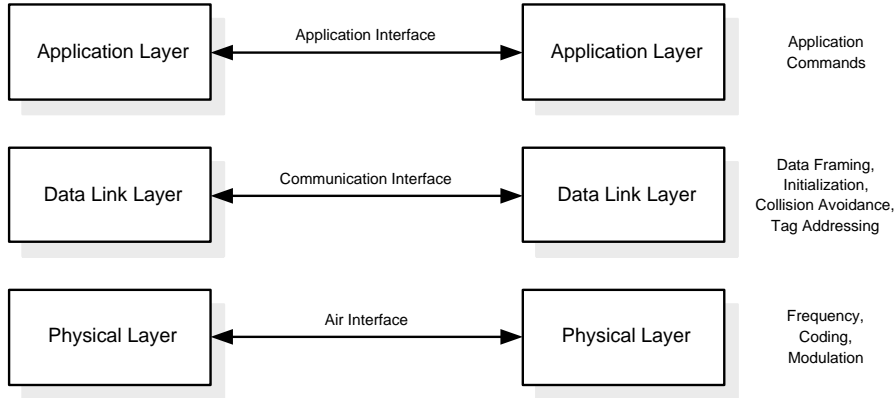


Figure 3.3: RFID Communication Model

Figure 3.3 shows the abstracted RFID communication model. The RFID communication protocol is separated into different layers. The upper layer consists of the commands issued by applications. These commands along with the data are put in frames or units in the intermediate layer. This layer also takes care of anti-collision, authentication, addressing and other communication mechanisms. The lower layer represents the technological view where coding and modulation take place.

### 3.6 RFID Standards

Standards are documented agreements containing technical specifications or other precise criteria to be used consistently as rules, guidelines, or definitions of characteristics, to ensure that materials, products, processes and services are fit for purpose [Org96]. The exploitation of technology in the form of products and services, offering scope for widespread national and international use, requires the support of standards. They facilitate compatibility and interoperability among devices and applications.

The relevance of RFID to virtually every sector of industry, commerce and services where items and data are handled, makes appropriate standards necessary. In table 3.3 common applications of current RFID standards are listed.

The ISO 14443 standard describes functionality and parameters for proximity coupling cards. ISO 15693 deals with vicinity cards supporting ranges of up to 1m and ISO 18000 as well as the EPCglobal standard target item management applications.

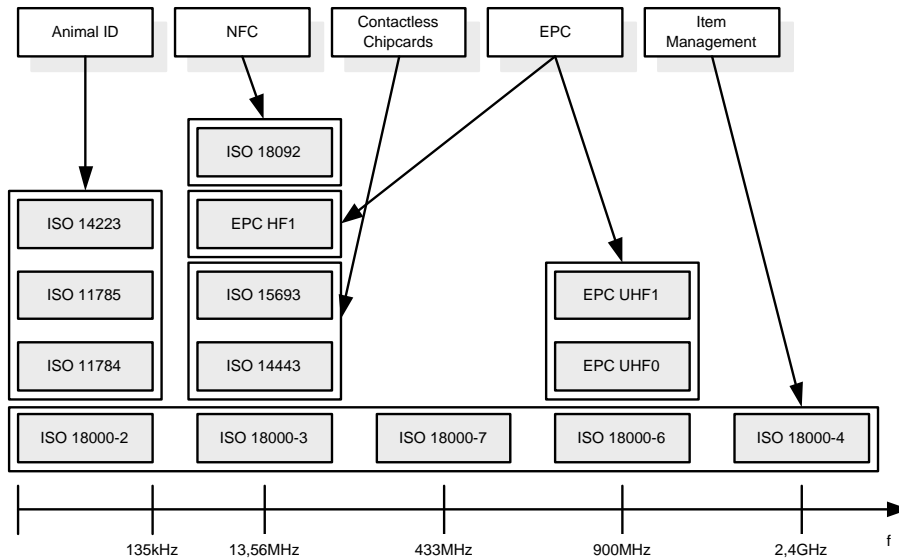


Figure 3.4: Overview of RFID Standards [KP04]

### 3.6.1 ISO/IEC 14443 Proximity Cards

The ISO standard 14443 describes function and parameters of proximity coupling cards (7-15cm) as used in ticketing or e-wallet applications. Most of these cards contain a microprocessor, which is the reason why they often come as dual interface cards.

Part 1 of the standard describes the physical characteristics of the cards. The measurements are the same as in ISO 7810 (cash cards): 85,72mm and 54,03mm. Furthermore part 1 contains regulations for the testing of bending load, torsional load and the exposure to UV- and electromagnetic waves.

In part 2 of the specification the radio frequency interface is defined. Power supply of the proximity integrated circuit card (PICC) is done via an alternating magnetic field emitted by the proximity coupling device (PCD) with a frequency of 13.56 MHz. In the communication interface two different types exist: type A and type B. Cards must support at least one of the two standards. A ISO 14443 conforming reader however has to support both types. This requires periodic switching between the two modes during idle state.

Part 3 covers the initialization and anti-collision mechanisms. If a PICC enters the field of a reader and a connection should be established, two things have to be considered. First there might be more than 1 card in the field and secondly there might already be an established connection with another card.

Part 4 of the 14443 specification covers the transmission protocol. The protocol is very similar to the ISO 7816-3 standard, which facilitates the construction of dual interface cards. It describes the transmission of application data units (APDU) which can contain



Application	Standards
Supply chain and parcel tracking	ISO 15693
	ISO 18000
Libraries and inventory management	ISO 14443
	ISO 15693
	ISO 18000
Smart passports and visas	ISO 14443
	ISO 18000
Airport baggage tags	ISO 15693 (UHF)
	ISO 18000
Smart cards and envelopes	ISO 14443
	ISO 15693
	ISO 18000

Table 3.3: Applications of RFID Standards

any data such as commands and responses. Data transmission can be described by the OSI layer model. Every layer carries out its tasks autonomously and is transparent to the upper layer. Layer 1 (physical layer) describes the transmission layer and the byte coding of data. The transport layer (layer 2) controls the transmission of data with correct addressing of data blocks (CID), sequential transmission of longer blocks, time behavior as well as the handling of transmission errors. Layer 7 contains the application data as a command to the chip card or the response. Layer 7 is independent from the current operating mode, which can be contact or contactless. Layers 3 to 6 are only used in complex networks and are omitted in this standard.

After a chip card has been activated it waits for the first command from the reader. Communication is strictly based on the master/slave principle. The reader sends a command to the card, which then executes it and sends a response.

### 3.6.2 ISO/IEC 15693 Vicinity Cards

Vicinity Integrated Circuit Cards (VICC) are contactless cards for ranges up to 1m such as they are used for access control applications. The reader is often referred to as Vicinity Coupling Device (VCD). Tags are commonly low cost with a simple state machine.

Part 1 (physical characteristics) defines the mechanical properties of vicinity coupling smart cards.

Part 2 (radio frequency power and signal interface) describes the radio transmission parameters. These include details describing the radio carrier frequencies, type and percentage of modulation, power levels, data rates, data encoding algorithms and system timing. These parameters form the communication protocol operating between a card and its reader. This protocol defines the data link for the air interface.

Part 3 (anti-collision) deals with the communication protocols necessary to allow multiple transponders to operate in a common excitation field. Each transponder is assigned a unique identifier (UID), an application family identifier (AFI) and a data storage format identifier (DSFID). As numerous transponders are detected in the field, each carries its own personal attributes and mission defined by its coding.

Section three also defines the complete communication interchange for requests and responses. A complete command set defines reading, writing and locking of data for single and multiple blocks. A CRC (cyclic redundancy check) block is included to assure the integrity of the data received.

### **3.6.3 ISO/IEC 18000**

There exists a whole range of new standards on the subject of item management. The ISO 18000 standard currently consists of seven parts defining the air interface. The first part explains how the standard works and the remaining parts deal with the various frequencies.

Part 1 describes the conceptual system architecture of RFID for item management and defines a common set of parameters that are necessary to avoid contention or interference with other RFID systems, to establish a high degree of interoperability and to ease integration with legacy systems. The subsequent parts of the standard provide the specific definitions of each of the approved frequencies, and where appropriate, provide regional definitions with geographical constraints.

In part 2 the air interface for radio-frequency identification (RFID) devices operating below 135 kHz used in item management applications is defined. Part 2 further defines the communications protocol used in the air interface.

Part 3 describes two different modes of operation, which are not interoperable although they are designed not to interfere with each other. Mode 1 is based on ISO 15693 with improvements, whereas Mode 2 defines a new, high speed interface. It relates solely to systems operating at 13,56 MHz.

The air interface for radio-frequency identification (RFID) devices operating in the 2,45 GHz band used in item management applications is defined in part 4 . The standard contains two modes. In the first mode passive tags are operating according to the "interrogator talks first" principle while in the second mode battery assisted tags operate according to the "tag talks first" principle.

Part 5 of ISO/IEC 18000 has been withdrawn, due to a lack of global acceptance.

In part 6 the air interface for radio-frequency identification (RFID) devices operating in the 860 MHz to 960 MHz band used in item management applications is defined. It contains one mode with two types. Both types use a common return link and are based on the reader talks first principle. Type A uses Pulse Interval Encoding (PIE) in the forward link and an adaptive ALOHA collision arbitration algorithm. Type B uses Manchester Coding in the forward link and an adaptive binary tree collision arbitration algorithm.

Part7 defines the air interface for radio-frequency identification (RFID) devices operating as an active RF Tag in the 433 MHz band used in item management applications.

It further defines the communication protocol used for the air interface.

Part	Specification
Part 1	Generic parameters for air interface communication for globally accepted frequencies
Part 2	Parameters for air interface communication below 125kHz
Part 3	Parameters for air interface communication at 13.56MHz
Part 4	Parameters for air interface communication at 2.45GHz
Part 5	Parameters for air interface communication at 5.8GHz (subsequently withdrawn due to lack of global acceptance)
Part 6	Parameters for air interface communication at 860 and 930MHz
Part 7	Parameters for air interface communication at 433MHz (late submission)

Table 3.4: Parts of the ISO/IEC 18000 Standard

The ISO/IEC 18000 series of standards deal only with the air interface protocol and are not concerned with data content or the physical implementation of the tags and readers. They are however designed to be capable of carrying EPC data.

### 3.6.4 Electronic Product Code (EPC)

EPC Global defines 5 classes of RFID tags, according to their read and write capabilities.

Class 0 tags are the simplest type of tags, where the data, which are usually a simple ID number (EPC), are written into the tag only once during manufacture. No further updates are possible. Class 0 is also used to define a category of tags for electronic article surveillance, called EAS. These tags have no ID and only announce their presence when passing through an antenna field.

Class 1 tags are manufactured with no data written into the memory. Data can then either be written by the tag manufacturer or by the user, but only once. After this no further update is possible and the tag can only be read. Tags of this type are usually used as simple identifiers.

Class 2 tags allow users to both, read and write data into the tag's memory. They are typically used as data loggers, and therefore contain more memory space than tags which carry only simple ID numbers.

Class 3 tags are just like class 2 tags except that they contain on-board sensors for recording parameters like temperature and pressure, which are recorded into the tags memory. As sensor readings must be loaded into memory in absence of the reader, the tags are either semi-passive or active, thus requiring an on-board power source.

Class 4 are equipped with integrated transmitters. These tags are similar to radio devices, which can communicate with other tags and devices in the absence of a reader.

Class	Specification
Class 0	"Read only" tags
Class 1	"Write once, read many" tags
Class 1 Gen 2	"Write once, read many" tags, UHF Gen 2 protocol
Class 2	Rewritable tags
Class 3	Semi-passive tags
Class 4	Active tags
Class 5	Readers

Table 3.5: EPC Classes

Presently deployed Gen 1 UHF RFID systems are based on a number of competing protocols, most notably Matric's Class 0 and Alien Technology's Class 1. There is a problem that these protocols are proprietary. Beyond that, they lack the features, reliability and power to adequately serve a growing number of applications, particularly when taking worldwide operability into account. MIT's Auto-ID Center recognized these problems and created a single open standard that would firstly create an environment of interoperability and international regulatory compliance and secondly would raise the bar on RFID system performance in a significant way. These two values formed the backbone of the EPC Gen 2 UHF standard. With a single worldwide specification in place, UHF RFID-based systems are expected to become faster, easier to use, less costly to deploy and more robust.

## 3.7 Proprietary Systems

Next to the open standards described in the previous section several companies have developed their own proprietary standards. These include My-d by Infineon, Mifare by Philips and FeliCa by Sony. Sony has cooperated with Philips to bring their FeliCa standard to Europe with FeliCa compliant Near Field Communication (NFC). Most proprietary systems target security applications and have some similarities in their concept. As an example the Infineon My-d system is described.

### 3.7.1 Infineon's My-d

My-d [Tec04] is based on the ISO15693 standard and can be either used in plain or in secure mode. In plain mode tags can be permanently locked against writing. Subsequent unlocking is impossible. My-d secure allows users the partitioning of the memory into sectors and the assigning of individual access rights to a number of entities. These rights can be read access, read/write access and no access. Some other special access rights such as decrement counter and write once for special applications exist as well. In order to achieve this sector keys are stored in an unreadable key area of the tag. If a reader, which

is in possession of the correct keys, wants to access a sector, it needs to authenticate with the corresponding key in the tag memory. Therefore the keys must be stored in the reader as well as in the tag. Tags can be configured to use diversified keys. In this case the tag's UID or a dedicated page is encrypted with the reader's key and used as the tag key.

Authentication is done with the challenge response protocol, to ensure authenticity from both sides. In the first step the reader sends the page address of the authentication counter and the page address of the authentication key to the tag. Subsequently the tag challenges the reader by sending a random number and the data of the authentication counter page to the reader. In the second step the reader calculates the response and returns it to the tag. The authentication counter is decremented and a MAC is calculated over the data. From then on all communication is verified by MACs to prevent session hijacking.

### 3.8 Summary

Radio Frequency Identification (RFID) makes use of radio transmission to recognize, categorize, locate and track objects. RFID systems consist of readers, tags and a back-end application or database for storage and management of the collected data. The tags are attached to the products and can be read when they enter a reader's antenna field. Since the system uses radio waves, there is no need of contact or direct line of sight between reader and tags. Tags can be powered by the antenna field of the reader, an external field or by a battery. The main characteristics and benefits of RFID are:

- It provides a per-item identifier.
- No line-of-sight is required.
- RFID is resistant to harsh environments.
- Tags can be reprogrammed and reused.
- Tags can be read in groups.
- Security mechanisms are available.

Vital for the large scale deployment of RFID is the standardization process, which guarantees component compatibility and open competition of suppliers thus reducing prices and dependencies. Next to the ISO 14443 and the ISO 15693 standards the ISO 18000 standard and the EPC standard are the most important ones. Particularly EPC Gen2 UHF is the most likely candidate for use in EPC infrastructures whose large scale deployment is imminent.

## Chapter 4

# Security Aspects of RFID

In this chapter we analyze the security issues that arise with RFID. Firstly an introduction to security in the context of RFID and a reconsideration of the security issues with special regard to RFID is given. After that, possible fraud scenarios are identified, discussed and weighed with their probability. From this starting point, current security mechanisms for RFID systems are presented and assessed. Finally the fraud scenarios are reconsidered with the existing security mechanisms and their efficiency in mind and a risk analysis is performed.

### 4.1 Security in the Context of RFID

In order to be considered secure an RFID system has to satisfy all or a subset of the security services described in chapter 2. When the security requirements have been outlined, it is possible to identify threats. This section deals with general attacks as well as specific attacks that are possible.

#### 4.1.1 Security Services Revisited

Depending on the security requirements all or a subset of the security services, that were presented in chapter 2, need to be satisfied by an RFID system. The following paragraphs revisit them taking into account the special characteristics of RFID systems.

##### **Confidentiality**

The communication between readers and tags is usually not protected. Eavesdroppers may thus listen if they are in the immediate vicinity. The forward channel from the reader to the tag has a longer range and is more at risk than the backward channel [WSRE03]. Furthermore, the tags memory can be read if access control is not implemented.

## **Integrity**

Only high security systems use message authentication codes (MACs). In general the integrity of transmitted information cannot be assured. Checksums (CRCs) are often employed on the communication interface but protect only against random failures. Furthermore, the writable tag memory can be manipulated if access control is not implemented.

## **Availability**

Any RFID system can easily be disturbed by frequency jamming. But denial-of-service attacks are also possible on higher communication layers. The so called "blocker tag" [JRS03] exploits tag singulation (anti-collision) mechanisms to interrupt the communication of a reader with all or with specific tags.

## **Authenticity**

The authenticity of a tag is at risk since the unique identifier (UID) of a tag can be spoofed or manipulated. The tags are in general not tamper resistant. Likewise a reader can be spoofed in order to get access to a tag's data.

## **Anonymity**

The unique identifier can be used to trace a person or an object carrying a tag in time and space. This may not even be noticed by the traced person. The collected information can be merged and linked in order to generate a persons profile. A similar problem occurs in supply-chain applications, where undesired product scans are possible. The automated reading of tags permits the counting of objects (e.g. banknotes with attached tags) which may be undesired.

### **4.1.2 Fraud Scenarios in RFID Systems**

For RFID systems a great variety of fraud scenarios can be identified.

#### **General Attacks on RFID Systems**

RFID systems provide a way to get the virtual world merged with the real world of objects and items. Therefore the integrity of RFID systems depends on three relations:

1. The relation of the tag to the data stored on it.
2. The relation of the tag to the object which is identified.
3. The relation of tags to readers.

Evaluating these criteria for a functional system, the following general fraud scenarios can be identified:

**Modification of Data:** By unauthorized write access, the data stored on the tag can be modified. This attack is only effective if the identifier and security information such as keys remain unchanged. Otherwise this attack leads to denial of service. The attack is only possible if additional data next to the identifier are stored.

**Tag Spoofing:** In this attack an attacker gets access to a tag's identifier and uses it to feign the original tag. This can be achieved either by emulating or by cloning. The requirement of uniqueness of tags is no longer fulfilled.

**Deactivation of Tags:** The tag is made inoperative by executing a dedicated command or by physical intervention. Depending on the degree of deactivation the identity or the presence of the tag can no longer be determined.

**Removal of Tags:** Under physical influence a tag is removed from the object, which is identified and maybe brought in association with another object. This is analogous to the exchange of price labels and violates the rule that a tag identifies its object.

**Eavesdropping:** The communication between tag and reader over the air interface is intercepted, decoded and interpreted.

**Blocking:** By the use of a blocker tag the presence of a arbitrary number of tags is simulated. A blocker tag has to be adapted to the specific collision protocol used.

**Jamming:** The data exchange over the air interface can be jammed either actively (jammer) or passively (shielding). Due to the susceptibility of the air interface even cheap passive methods can have an effect.

**Reader Spoofing:** In a secure RFID system readers have to prove their authority to read out tags. When an attacker wants to get access to the data on the tag, he has to feign an authorized reader.

Figure 4.1 shows that there are various intentions an entity that attacks an RFID system might have. An attacker may want to access sensitive information or exploit an RFID system by spoofing an RFID tag. A saboteur's or terrorist's intention might be to make an RFID system unavailable (DoS attack). Even a user might launch an attack because he feels his right for privacy violated.

### **Attack Mechanisms on RFID Systems**

Attacks can either be targeted at the active party or the passive party. Potential attackers are the passive party (employees, customers) or a third party (terrorists, saboteurs, competing companies). The likelihood of attacks from different groups varies strongly. The usefulness of an attack by a competing company is relatively small compared to consequence of the loss of reputation in case of discovery.



	Protection of Privacy	Access to Data	Denial of Service	Spoofing
Modification of Data				
Tag Spoofing				
Deactivation of Tags				
Removal of Tags				
Eavesdropping				
Blocking				
Jamming				
Reader Spoofing				

Figure 4.1: Intentions behind Attacks on RFID Systems

**Accessing Data** If an attacker wants to get access to the (confidential data) stored on a tag he can choose one of the following ways:

1. The use of a compatible receiver enables an attacker to listen to the communication over the air interface. From distances greater than the specified one, communication can be eavesdropped partially [WSRE03].
2. With the use of a compatible reader the contents of a tag can be read. Because of the standardization of RFID equipment, compatible readers can easily be acquired. In case the reader has to authenticate itself, the attacker has to be able to spoof the identity of the reader.

**Spoofing** An attacker intending to spoof the active party has several options.

1. The attacker might change data stored on the tag but not the unique identifier. This is only possible if the data are stored on the tag and not in a back-end system, which is not necessary for most applications.
2. In order to spoof a tag, an attacker has to duplicate a tag either by emulation or by cloning. To do this, an attacker has to know at least the unique ID or passwords and keys depending on the security mechanism.
3. An attacker can remove the tag from its object destroying their association and possibly apply the tag to another object. Depending on the mechanical security mechanisms, this includes the destruction of the tag or the object limiting the effectiveness of this attack.

**Denying Service** Due to the susceptibility of the air interface, RFID systems are vulnerable to a great number of DoS attacks. Fortunately DoS attacks are only partially useful and easily detectable. Possible ways to cause denial of service are:

1. An attacker might chemically or mechanically destroy the tags.
2. Tags can be destroyed by electromagnetic radiation similar to the process of deactivating 1-bit transponders used for article surveillance. This can be achieved by specific transmitters, but also by microwaves and induction.
3. Also the unauthorized execution of a kill or delete command can be used to deactivate a tag. In order to do this an attacker has to be able to spoof an authorized reader.
4. In systems that use active tags, the battery can be depleted by sending a large number of queries to the tag.
5. Using a blocker tag the presence of an indeterminable number of tags can be simulated preventing the reader from accessing an individual tag.
6. With a jamming transmitter the communication over the air interface can be impeded. For greater distances very large transmitters are necessary. A jamming attack can easily be detected.
7. Reflecting objects can cause wave extinction.
8. Adjacent water and metal can impede correct operation.
9. By surrounding tags with metal, they can be shielded from reader fields.

## 4.2 Current RFID Security Mechanisms

Now that the security context for RFID systems has been established, various mechanisms for satisfying the security services are presented. The security considerations are based on the assumption that tags are secured against attacks that target the hardware directly. These attacks include side channel attacks such as timing attacks, differential and simple power analysis attacks (SPA, DPA) and differential and simple EM analysis attacks (SEMA, DEMA). As countermeasures hardware design techniques, that are beyond the scope of this paper, can be employed. Readers are considered to be secure against low level attacks as well. Keys are assumed to be stored in an area of the memory that is protected from reading and unauthorized writing.

### 4.2.1 Authentication of Tags and Readers

Authentication guarantees the authenticity of the reader, tag or both entities. This is to make sure that only entitled readers may execute commands on the tag and only data from entitled tags are accepted to be valid by the reader.

The authentication of readers prevents all attacks that include the execution of commands on tags by unauthorized entities. This can be the writing of inconsistent or fraudulent data on the tag as well as the initiating of read commands making the tag release valuable data. The authentication of tags prevents cloning and the creation of illegal tags containing wrong data.

There are several ways to implement authentication. This can be for example via a protocol which requires a shared secret, or two public private key pairs. Also passwords can be used as an authentication mechanism. Another option is to perform authentication at data level with digital signatures. Authentication has to be guaranteed during the lifetime of the communication that has to be protected. A violation of this rule can for example be caused by session hijacking.

### **Passwords-Based Authentication**

Passwords only provide weak authentication because the knowledge of a secret token suffices to get access. Systems are also susceptible to dictionary and social engineering attacks as well as replay attacks. Password based authentication is a reasonable choice for low cost read-only tags that only need to have specific commands secured. It is for example used for ISO18000 class 1 tags and for EPC tags to execute the kill command. Short passwords can be decoded with a few attempts, which is known as brute-force attack. Variable passwords provide better security but require R/W transponders because the password must be rewritten.

The ISO 18000 standard only allows for static passwords sent in the clear from reader to tag. The EPC Gen2 UHF standard covers the password with a random number sent from the tag to the reader to protect the forward link. Current deployments only employ write passwords. There are two approaches to password management:

1. Single password per site.
2. Unique password for each tag.

If a single password is used for all tags, a compromise of a tag poses a threat to the whole system. In deployments that use write passwords, the password is sent only when data are written, whereas in systems with read passwords, readers must use the password every time data are read. The danger of a passive eavesdropper is therefore higher in the second case. However, eavesdropping on a single communication channel reveals the password used by every tag in the system, a serious security failure. Once learned by a single adversary, the password can easily be made public. Then, anyone with a reader can use this password to get access to the tag.

If different passwords per tag are used, then a mechanism is required to allow the reader to determine which password should be used for which tag. This can be accomplished for example by having the tags send an index to a table of shared secrets to the reader. These mechanisms are complex and costly to implement and are therefore rarely used.

## **Authentication at Data Level**

Another method in conjunction with encryption is the storing of data on the chip in encrypted form. The data are encrypted by the application or the reader and written to the chip. This can only be done with read-only transponders because the execution of commands on the tag is still possible. However this way valuable data are protected from eavesdropper and no additional functionality of the tag is required. Apart from that no key has to be stored in the transponder. The transmission over the air interface is also secured from eavesdroppers since the data are never transmitted in plaintext.

For guaranteeing integrity data can be stored along with their digital signature to guarantee authenticity. Encryption with a public key is also possible in which case the data size increases.

## **Hash-Lock Authentication**

Hash-lock based authentication mechanisms provide better security than passwords in that longer key sizes may be used. Before issuing tags a meta-id, the hash of a key  $K$ , is calculated and stored in the tags. From then on the tag is locked, meaning that it replies to all requests with its meta-id. Only readers sending the correct key can get access to the tag's contents. When the tag receives a key, it calculates its hash and compares it with its meta-id. If they match, access is granted. The keys are either stored in entitled readers or in a back-end database.

Due to the longer key sizes, brute force attacks become infeasible and dictionary attacks can be avoided by choosing random numbers as keys. However the mechanism is still vulnerable to replay attacks because the key is statically transmitted over the air interface.

## **Challenge Response Authentication**

The process of authentication involves the proof of knowledge of a shared secret. In zero knowledge techniques an entity can prove its authenticity without requiring the other entity to know the secret or derived information of it. They usually consist of several rounds and calculations.

Challenge-response protocols can provide mutual authentication in two or three steps and unilateral authentication in one or two steps. Mutual two-way authentication and unilateral one-way authentication require the knowledge of a time-dependant value such as a timestamp or a timestamp seeded random number. Because this is difficult to implement, random numbers are preferred.

When public-key cryptography is used, the verifier sends a challenge to the signer, which the signer encrypts with his private key. The verifier then decrypts the message with the signer's public key. When the result is the same as the sent data, the signer's identity is proved. The advantage is that every entity has its own private key which doesn't have to be shared. Common public key infrastructure techniques and practices can be used to manage tags and readers. The major drawback that makes public key systems

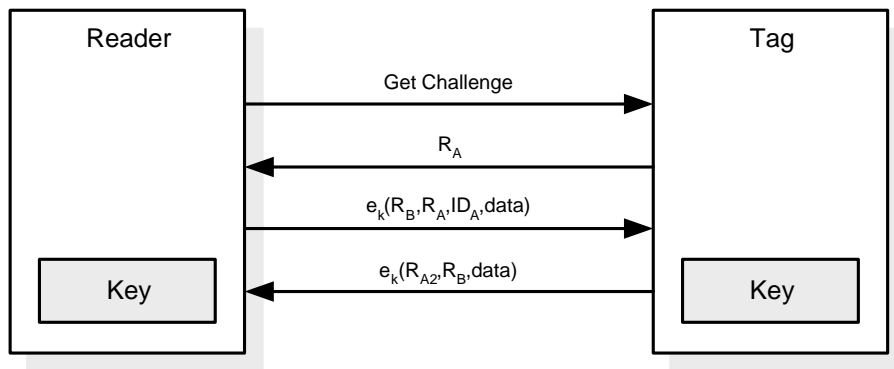


Figure 4.2: Challenge-Response Authentication

unsuitable for RFID applications, is that algorithms are computationally too intensive without a processor. Furthermore power is limited, so that only high end systems can take advantage of it. One such system was developed by the Number Theory Research Unit (NTRU) group and is called GenuID (see [NTR02] for further information).

Symmetric key cryptography is much faster, but a secure mechanism for key distribution and policies for the case of reader or tag compromise have to be thought of. As an algorithm the 128 bit AES (advanced encryption standard) is suggested because of standardization, ease of implementation and better performance compared to 3DES. AES supports key sizes of 128, 192 and 256 bits.

The advantages of mutual authentication are:

1. The keys are never transmitted during communication.
2. Two random numbers are always encrypted simultaneously. This rules out an inverse transformation using  $R_A$  to obtain token 1.
3. Tokens can be encrypted with any algorithm.
4. The strict use of random numbers from two independent sources prevents replay attacks.
5. A session key can be calculated from the two random numbers in order to secure the subsequent communication with symmetric cryptography.

**Unilateral Authentication** For read-only tags, where the data are not of value, authentication of the reader is not necessary because no harm can be done by writing on the tags and the data are useless to the attacker. This is the case if the tag only contains an identifier and the data related with it is stored in a secure back-end database, or if the data are stored in an encrypted format on the tag, or the data are simply unimportant. An

authentication of the tag however is necessary to rule out the use of fake tags. Examples for use cases of read-only tags where unilateral authentication is required are ticketing, item management (EPC, ISO 18000) or automation applications, where machines rely on data from tags.

RFID applications operate on the master-slave principle. This means that all communication is initiated by the reader.

$$\begin{aligned} R &\rightarrow T : r_R \\ T &\rightarrow R : E_K(r_T, r_R) \end{aligned} \tag{4.1}$$

In step number 1 the reader sends a random number (to prevent replay attacks) to the selected tag which is to be authenticated. The tag encrypts the random number and sends it back to the reader, which verifies it. This process is illustrated in figure 4.2.

**Mutual Authentication** Whenever a tag supports read/write functionality or is read-only but contains valuable data, an authentication of the reader is required. Also the reader must be sure about the authenticity of the tag, making mutual authentication necessary.

Authentication is done in three steps:

$$\begin{aligned} R &\rightarrow T : r_R \\ T &\rightarrow R : E_K(r_R, r_T) \\ R &\rightarrow T : r_T \end{aligned} \tag{4.2}$$

The reader sends a random number to the tag which encrypts it along with its own random number and sends the result back to the reader. The reader verifies the random number and sends the decrypted random number back to the tag which can then verify the authenticity of the reader the same way.

**Challenge-Response Authentication with Hashes** In low cost products the implementation of a cryptographic processor might not be cost effective. A solution is to use a keyed hash function to encrypt the random number and append it as a signature. The hash value will only be a few bits long (usually 32) and is therefore susceptible to brute force attacks. Consequently the number of authentication attempts should be limited by using an authentication counter. The security equivalent should be that of a 128 bit key.

$$\begin{aligned} T &\rightarrow R : r_T, Ctr \\ R &\rightarrow T : r_R, (Ctr - 1), H_K(r_R, (Ctr - 1), r_T) \\ T &\rightarrow R : (Ctr - 1), H_K((Ctr - 1), r_R) \end{aligned} \tag{4.3}$$

Several light-weight authentication protocols for low-cost systems are presented and analyzed in [VB03].

**Continuity of Authentication** Once a security association between a reader and a tag has been established, continuous authentication has to be guaranteed. Otherwise a tag or reader could be replaced during communication without one of the parties noticing it (session-hijacking). Re-authentication is not suitable because it would slow down communication and even the smallest unauthenticated time-slots can be exploited. An acceptable mechanism is the use of keyed hashes where a hash is calculated over the data  $D$  and the key and appended to the data to be sent as can be seen in equation 4.5. Each entity verifies the hash before accepting the data.

$$T \rightarrow R : D, H_K(D) \tag{4.4}$$

$$R \rightarrow T : D, H_K(D) \tag{4.5}$$

The length of the keyed hash has to be chosen carefully to balance security and communication efficiency. Pre-image resistance is not so important because only a limited number of commands exist. Thus a second pre-image is practically useless to an attacker. Another option for guaranteeing continuous authentication is the negotiation of a session key. This option is computationally more intensive but can substitute the use of a hash, when encryption over the air interface is required (see section 4.2.2).

**Hierarchical Authentication** In some applications tags are used in various environments and use cases. For example an employee card might contain a key pointing to data in a back-end database, contain access data, and a temporary money cache that can be filled. Access data are plainly readable but can only be written by entitled readers from the security staff. The money cache can be read and rewritten by the corporate restaurant but by nobody else. In this case the memory is split into sectors, each one protected by a key pair with each key having different access rights. Instead of authenticating the card each sector has to be authenticated separately. One byte in each page is reserved for access control data for the key pairs.

## 4.2.2 Encryption

Passive RFID tags receive all their power from the RF field of the reader. Therefore there are some limitations on the power consumption of microprocessors used on the tag. High-end cryptographic processors, as they are used on contact cards and dual interface cards, are not easily integrated in the hardware design. Rising costs lead to a loss of competitive edge for the product although read/write tags using authentication (which is recommended) always have to implement an encryption algorithm or a keyed hash function. However these algorithms only have to support encryption and not decryption. In [Jue04] a minimal cryptography scheme for low cost RFID tags is presented.

## Overview

The tags that will be most inexpensive and most prevalent, such as basic EPC tags, lack the computing power to perform even basic cryptographic operations. They will have about 500-5000 gates, many devoted to the basic tag functions. By contrast, the Advanced Encryption Standard (AES) requires some 20,000-30,000 gates. Such tags are at best capable of employing static keys such as PINs and passwords as security mechanisms. For example, the "kill codes" used to disable EPC tags for purposes of privacy, are secured by PINs. The limited capabilities of such RFID tags make privacy and security enforcement a special challenge.

More expensive RFID tags are capable of advanced functionality and often include the ability to perform basic cryptographic algorithms, such as symmetric-key encryption and challenge-response identification protocols. Public-key cryptographic is expensive and used on few, mainly active RFID tags.

Encryption guarantees the privacy of the data by making the information unreadable for eavesdroppers. Consequently encryption is mandatory when the data must be protected. It is all the more important when any reader may execute commands on the tag. Read/write tags are additionally secured from unauthorized reading by an authentication mechanism. However when the air interface is susceptible to eavesdropping, as it is the case with long range systems, data should be transmitted in encrypted form.

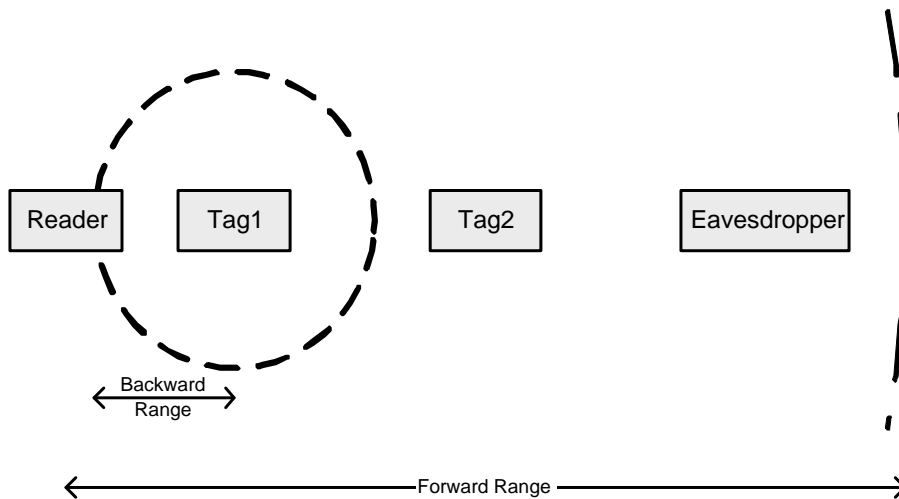


Figure 4.3: Eavesdropping on Reader-Tag Communication

In figure 4.3 a typical case of eavesdropping is shown. The eavesdropper is out of range of the tag signal and can only listen to the commands issued by the reader. Tag 2 is not recognized by the reader.



## Encryption over the Air Interface

Encryption over the air interface requires both, the tag and the reader, to implement a cryptographic processor. Data is sent plainly to reader and is there encrypted with a standard algorithm (3DES, AES). Then the data are sent over the air interface to the tag which decrypts the data and stores it in its memory. If data are sent from the tag to the reader, the data are encrypted in the tag, and sent to the reader where they are decrypted again. The reader can then forward the data plainly or encrypt them again for network transport.

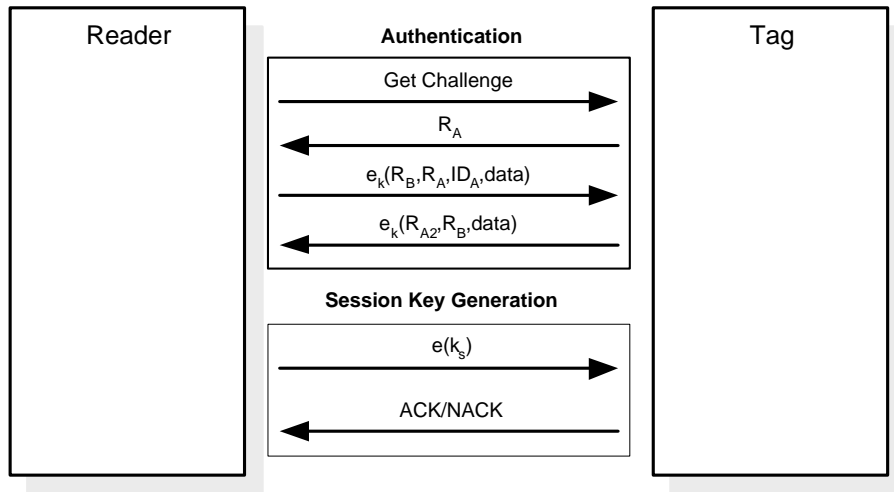


Figure 4.4: Session Key Selection

In this procedure the selection of the session key is taken care of by the reader or the application once both entities have been authenticated. During transfer the session key is encrypted with the authentication key. From then on the session key is used to encrypt data over the air interface. Reader as well as tag have to implement a symmetric cipher, supporting encryption as well as decryption. The advanced encryption standard allows for some components used in the encryption to be reused by the decryption process.

## Encryption at the Data Link Layer

When an authentication mechanism is used, the tag as well as the reader have to implement a cryptographic algorithm. However carrying out encryption in the tag is inefficient and would significantly delay the tag's responses. Usually it is a better choice to do encryption in the reader, which results in little or no performance hits of the overall system. Communication from the reader to the application is then secured with a network optimized security scheme. The algorithm in the reader can either be implemented in hardware, if a

great number of readers is used, or in software. As a key for encryption the authentication key or a separate key may be used. The drawback of this solution is that it complicates key management. Furthermore the compromise of a reader poses a security threat in that it may read and write all transponders encrypted with the used key. This necessitates a key exchange for all readers, which can be a costly procedure. On the other hand this mechanism relieves the application programmer from taking care of encryption.

### **Encryption at the Application Layer**

Unilateral encryption can also be put in the hands of the application. This gives the programmer the opportunity to select an encryption scheme that may better fit a specific application. It might not even be necessary to use encryption at all, except for specific data. The scheme allows for more fine grained handling of data. For example an application may only encrypt vital data while leaving the rest of the data plain. This approach for encryption reduces costs and provides scalability. It also eliminates the need for encryption over the network.

## **4.3 Additional Considerations**

When implementing security mechanisms in RFID systems, some additional aspects have to be taken into account. Particularly important is the key management, which, when not carefully planned, poses a significant security risk. Diversified keys can help make systems more secure by eliminating threats that result from key compromise in the tag. Another important issue is the support of tags for various applications with different access conditions.

### **4.3.1 Key Management**

A key management system ideally provides transparent and centralistic operation. Each entity must be addressable by a unique identifier such as a serial number or a computed hash value for keys. The key management is either integrated in the RFID middleware or takes place in a dedicated area (trust center). In the first case high security measures have to be implemented in the system to prevent the compromise of keys.

#### **Key Initialization**

The process of writing keys into the reader and the tags has to be secured by additional "key transfer keys". For tags this key is called the chip transfer key  $TK_C$  and for readers it is called the reader transfer key  $TK_R$ . These keys should be changed from their default value after delivery from the manufacturer.

For the transfer of keys into the reader, the reader has to be connected to the key management system. Prior to transfer the key management system authenticates to the reader with  $TK_R$ . Optionally the keys may be symmetrically encrypted with  $TK_R$  during

transmission to prevent eavesdropping and to comply with the policy that keys are never readable.

When readers are no longer used they should be reset, meaning all keys are deleted. The transport key should be set back to the default value. For this step to succeed, it is also required to have the correct  $TK_R$ .

Some systems such as Infineon's my-d support key transfer with a key transfer transponder. This way keys can be conveniently loaded into the reader by putting a key transfer transponder in its field. The reader has to be put in a special mode called the key transfer transponder awareness mode.

Transponder issuing is carried out by dedicated readers that contain all the necessary keys and have the right to write keys into tags. The keys are held in a secure key store and cannot be read from outside. The reader authenticates to the tag with  $TK_C$  diversifies the tag's UID and writes it into the tag's read-protected memory.

### 4.3.2 Diversified Keys

The tags used in an RFID application are usually not in full control of the system owner. For example tags are issued to passengers in ticketing applications, or bank cards are issued to the customers of a bank. From then on it is within their responsibility to take care of the cards. This in turn would mean that using symmetric keys, every card has to use the same key or every SAM has to store a different key for every tag. The first option is not suitable because the compromise of a single tag results in a threat to the system. Using the second option, key management becomes a real hassle. A good solution is the use of diversified keys.

$$k_d = e_{k_m}(UID_{tag}) \quad (4.6)$$

For the creation of a diversified key  $k_d$  each entity must have a unique identifier  $UID_{tag}$ . The UID is encrypted with a master key  $k_m$  (see equation 4.6) and stored in a safe container on the tag. For RFID applications usually the serial number, which is also used to address the tag for communication, is chosen as the unique identifier. It is encrypted by the reader's master key and stored in a safe memory area on the tag.

During authentication the reader requests the UID from the tag and encrypts it to get the tag's key as can be seen in figure 4.5. For further communication mechanisms where a key is involved (authentication, encryption), the diversified key  $k_d$  is used.

The use of diversified keys facilitates key management in that it ensures that each tag has a unique key and in that readers only have to store a single master key. Domains can be built when not the serial number but a different random number for each domain is chosen. This however should only be done with hierarchical authentication mechanisms on higher levels. For basic card authentication unique keys should be used, allowing the blacklisting of tags in case of a loss. The master key must be carefully protected because it enables communication with all tags in the domain.

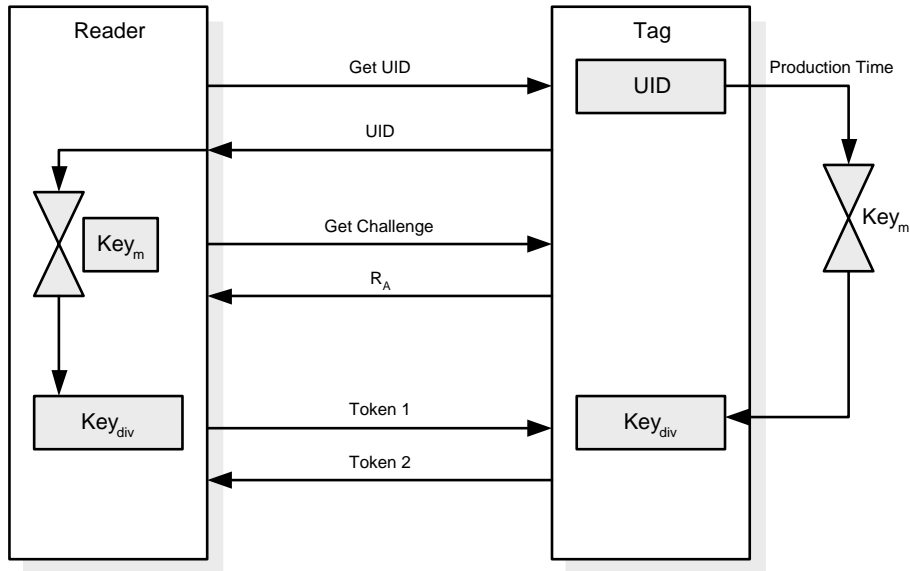


Figure 4.5: Key Diversification

### 4.3.3 Multi-Application Support

Some applications need to be used in several domains for which different access rights can be set. In this case the use of several keys is necessary and a mechanism for identifying and locating the designated key is required. Systems that are designed to support multiple applications are for example Infineon's My-d (see [Tec04]) and DESFire from Philips (see [Sem03]). The tag's memory is partitioned into sectors for which different access rights for the keys are set. The access rights are:

1. Read-Only
2. Read/Write
3. No access
4. Write-Only (for keys)

The keys are stored in a dedicated area of the tag that only supports writing and is protected by a master key. The same applies for the readers. Each sector grants certain access rights to certain keys. Whenever a user wants to access a specific sector he has to authenticate it with the correct key. Each sector is identified by a sector index, which is mapped to two pages in the memory area dedicated to the storage of keys. These two pages contain two keys: one for authenticating for read/write access and one for authenticating for read-only access.

To locate the physical address of a specific sector and the corresponding keys dynamically, a data structure has to be implemented on the tag. This data structure is called the sector allocation table (SAT) and maps the sector identifier to a physical start address in memory and the two key pages used for authentication. This approach also guarantees scalability because the adding of a sector only requires the allocation of new memory for the sector, the writing of the keys and a new entry to the SAT. Another advantage is the support for hierarchical key management, which can be implemented by creating a sector to hold the assigned keys and protecting it with a master key. The locating of the SAT is done by a pointer which is part of the project identifier page. This page can be freely accessed to decide whether the tag is part of the project. A related technology is the Mifare Application Directory (see [Gro03]).

The key management for authentication can be assigned to a dedicated microprocessor, called the security access module (SAM). A slot has to be provided by the reader to which the SAM is attached. The SAM can be individually programmed for a specific application and can then support the management of keys associated with the tag communication as well as with the communication to the middleware. A drawback of this approach is that the hardware causes additional costs.

The software approach to managing keys in the reader is a database which stores hash values of the keys, a link to the memory sections they provide access to, and which kind of access (RO, RW) they provide. This database approach transfers the responsibility of carrying out security functions to the middleware, but they can be automated by providing a security API. Now if a certain sector needs to be authenticated for a writing task, the corresponding key address in the reader has to be looked up in the database and the key address in the tag has to be read from the tag's SAT. If both keys exist and are valid, the authentication is carried out, else an error code is returned. When tags and readers are personalized and issued, details about the data model and security structure are known. The information may be stored in a database and can be used to generate runtime information.

## 4.4 Security Threats Evaluated

When evaluating the security risks to RFID systems in the medium and long term, it is important to consider the costs an attacker has to spend as well the costs and efficiency of countermeasures. Rising fixed and variable costs with additional security mechanisms can be justified when a great number of pieces are produced.

### **Eavesdropping on the Air Interface**

Without further security measures it is possible to eavesdrop on the air interface. The risk increases with the maximum reading range. For tags with a very short range the risk is relatively low. In [fSidI04] it is estimated that for inductive coupled systems it is possible to eavesdrop on the downlink up to 10m. For the uplink it is about five times the

intended range. In backscatter systems at 2mW power, the downlink can be tapped as far as 100-200m. With a beam antenna tapping at even 500m to 1km is possible. However a problem with eavesdropping at greater distances is the determination of the source of the signal.

The costs for an attacker are high because he needs professional equipment and know-how in the decoding of the data. Even though most communication over the air interface is standardized building a device for eavesdropping requires expert skills.

Countermeasures include:

1. Shifting data into the back-end and storing only an identifier. This approach also facilitates data management.
2. Shielding zones where readers are used.
3. Encryption of communication over the air interface.

### **Unauthorized Access**

To gain unauthorized access to a transponder, an attacker needs a stealth reader. This is not hard to accomplish, since most RFID systems comply to one of the currently existing standards. In monitored areas attacks of this kind can be effectively prevented because range is limited. The construction of readers with longer ranges than the standardized one, requires expert knowledge. For inductive coupling systems, range can be doubled at considerable costs, with an upper limit of 1m [fSidi04]. In the UHF range, power is limited to 2W by law. With 30W ranges around 10m can be achieved. For 20m more than 500W are necessary. These power levels already lying in the area of radio broadcast stations are unsuitable for stealth operation. Reading distance is also limited by the fact that the weak tag signal is superposed by the reader signal. Many RFID systems are designed to allow reading only from a very short distance (banknotes, chip cards).

For read-only tags data tampering can be ruled out. Read/write tags, on the other hand, are additionally vulnerable to attacks that include the modification of data. Countermeasures include:

1. Shifting data into the backend and storing only an identifier.
2. Detectors that can recognize readers.
3. Authentication.

Detectors raise costs only minimally when great areas can be covered with few detectors. Authentication usually has a much greater impact on the price per unit compared to simple read-only tags.

## **Cloning and Emulation**

Cloning is the creation of a new tag with data, that was previously obtained from a valid tag. This tag can then be used to spoof the identity of the original tag. Apart from that a device can be used to emulate any arbitrary tag (emulation). This way a smart shelf system can be fooled by replacing an item's tag with a clone or an emulator. Countermeasures include:

1. All countermeasures against unauthorized reading.
2. Authentication.

## **Removal of Tags**

Removal of tags is a trivial attack that can be easily carried out without equipment. Attacks of this kind target the relation between the tags and the item they identify. Intentions behind the removal of tags can be fraud or tampering. The risk of tampering is particularly high when it can be carried out easily and without detection. Possible countermeasures are:

1. Close mechanical link between tag and item.
2. Hiding tags in item.
3. Active tags with alarm function.
4. Additional identifiers (barcodes, watermarks).

## **Destruction of Tags**

A destruction of tags can be accomplished mechanically or with an electromagnetic field. For mechanical destruction the same security measures as for the removal of tags can be used.

Destruction with electromagnetic fields is based on the same effect that is used in EAS. Even though a deactivation prior to purchase can be carried out relatively easy in current EAS methods, no cases have been reported so far. Usually the induced voltage in a tag is limited by zener diodes or stabilizers. However, when voltage exceeds the load limit, the tag can be destroyed. Attacks are only effective when carried out from proximity ruling out the risk of a mass destruction of tags. Currently there exist no practicable countermeasures.

If tags are equipped with a kill command, an abuse thereof can also lead to an unwanted destruction of a tag. Therefore kill commands should be secured by authentication mechanisms to prevent unauthorized execution.

## **Blocking**

In contrast to jamming the use of passive blocker tags is not prohibited by law. The available blocker tag from RSA is only applicable for RFID systems that use the tree-walking algorithm for anti-collision. For different protocols new blocker tags have to be designed and created, which is technically possible. So far there exist no technical countermeasures against blocker tags. The only way to prevent (legal) blocking is a prohibition in the terms and conditions of the system operator.

## **Jamming**

Effective jamming from greater distances requires strong transmitters, whose unlicensed operation is illegal. For the general public it is hard to get access to the necessary equipment. Available countermeasures include:

1. Scanning for jammers.
2. Frequency hopping.

## **Shielding**

Shielding can be carried out by wrapping the tag in a metallic foil. The problem with shielding is that it is relatively easy to do but impossible to rule out completely. Still, antennas from different directions and better readers may solve the problem partially.

## **4.5 Summary**

Depending on the requirements RFID systems need to satisfy the security services or a subset thereof. Special characteristics of RFID systems are that their integrity depends on the relation between tag and data, tag and object as well as tag and reader. This allows the following fraud scenarios to be identified: Modification of data, tag spoofing, deactivation of tags, removal of tags, eavesdropping, blocking, jamming and reader spoofing.

With a properly implemented authentication algorithm it is no longer possible for attackers to execute commands on tags or to feed wrong and unauthenticated data to a reader. A danger however is the compromise of a key. This way an attacker could produce a forged reader that could be used to read and write tags or forge tags that are accepted to be valid by secure readers. The forging of tags is complicated by using diversified keys. Keys should be written to the transponders and to the reader in a secure environment. Brute force attacks on the authentication algorithm are infeasible if the encrypted random number is long enough or the number of authentication attempts is limited by an authentication counter. The case of reader compromise necessitates the exchange of all keys on readers and tags because with this reader would enable an attacker to authenticate to a tag and carry out commands on it.



Encryption prevents eavesdropping which is particularly important for long range systems and whenever read commands may be carried out by any reader. A potential attacker trying to eavesdrop on the data exchanged between tag and reader only reads encrypted data which are useless to him unless he knows the keys. The session key can either be derived from the random numbers exchanged in the authentication process or provided by the reader over an encrypted channel. The danger of eavesdropping only exists for systems operating at longer ranges and therefore only those systems have to be secured in such a way. If systems operating at shorter ranges have to be secured against eavesdropping depends on the importance of data stored on the tag. Though highly unlikely, an attacker might for example install a rogue reader in a way unrecognizable to users. In most cases it usually suffices to not store the data directly on the tag, but move them to a back-end database. Then the tag only stores an identifier, which can be a serial number, providing no relevant information to a potential eavesdropper. This is the approach that is used in EPC systems [EPC05]. Another advantage of this approach is that storage space is not limited compared to tags providing only a few kilobytes. Insignificant data such as they are used in production processes usually do not have to be protected from eavesdropping. Instead a locking mechanism (hash lock) can be used to prevent sabotage by writing invalid data to tags. However an attacker, whose goal is to sabotage a system, would rather launch a DoS attack which is much easier to carry out.

Security risks to RFID systems in the medium and long term depend on the costs an attacker has to spent as well as the costs and efficiency of countermeasures. Table 4.1 lists attacks and possible countermeasures.

Attack	Costs	Countermeasures	Costs
Eavesdropping on the Air Interface	high	Data in Backend Shielding Encryption	medium
Unauthorized Reading	medium-high	Detectors Authentication	medium
Data Tampering	medium-high	Read-Only Tags Detectors Authentication	low-medium
Cloning and Emulation	medium	Duplicate detection Authentication	medium
Removal of Tags	low	Secure attaching Active tags with alarm Additional identifiers (barcodes)	low-medium
Mechanical or chemical destruction	low	Secure attaching	low-medium
Destruction with electromagnetic fields	medium	none	
Destruction with kill command	medium	Authentication	medium
Blocker Tag	low	none	
Jammer	medium-high	Detectors Frequency Hopping	medium-high
Shielding	low	Better readers (only partially effective)	medium

Table 4.1: Attacks on RFID Systems and Countermeasures

## Chapter 5

# RFID and Privacy

Alongside with security issues, privacy is another important factor that needs to be taken into account. This chapter presents the threats to privacy posed by RFID in various scenarios. As in the previous chapter first current methods for guaranteeing privacy are presented. Then the privacy threats are reevaluated and analyzed with consideration of the mechanisms available.

### 5.1 Threats to RFID Privacy

An RFID system involves two parties with different interests. On one side, there is the system operator called the active party. The active party is in full control of the data and their use. It also issues tags and manages the data associated. On the other side, there are the employees and customers, referred to as the passive party. Usually the passive party has no influence on how the data on the tag is used.

For the active party the correct function of the RFID system is vital, whereas for the passive party it is important that the advantages of the new technology outweigh its disadvantages. Especially consumer protection organizations fear that RFID systems could interfere with the right on privacy. This made Simson Garfinkel state an RFID Bill of Rights [Gar02]:

1. The right to know whether products contain RFID tags.
2. The right to have RFID tags removed or deactivated when they purchase products.
3. The right to use RFID-enabled services without RFID tags.
4. The right to access an RFID tag's stored data.
5. The right to know when, where and why the tags are being read.

Privacy is a threat that primarily affects the passive party, which comprises for example customers and employees in RFID systems. The passive party uses either tags or objects

that are identified by tags but has no control of the data stored on the tags. Privacy can be violated by the active party or a third party. Since the active party is in full control of the data obviously no attack is necessary in the first case leaving only third party attackers. Nevertheless the active party can violate data protection laws by handing over private data to third parties. These scenario is not an issue of RFID security and is not dealt with in this thesis. In the second case a third party directly tries to gain access to privacy relevant data. Consequences for the passive party are similar because data is transferred without consent and knowledge.

## 5.2 Privacy Scenarios

Depending on the privacy right concerned it can be distinguished between data privacy and location privacy. Both are equally important to consider.

### 5.2.1 Data Privacy

If an RFID system stores personal data the privacy of the passive party is threatened in the following ways:

1. By eavesdropping on the air interface or unauthorized reading of tags an attacker can gain access to personal data.
2. Next to personal data also potentially personal data can be the target of an attack on privacy. This includes data that are anonymized but can with high probability be dereferenced.
3. The high congruency between the real and the virtual world can awaken the interest of other parties in the data. It might for example be possible that the police legally enforces access to the collected data, which might not be wanted by the users.

### 5.2.2 Location Privacy

Usually the passive party is in possession of RFID tags for a longer period of time. As a consequence an attacker is able to create a movement profile of the victim by reading out the static identifier on a regular basis. This is referred to as tracking. The danger of tracking rises when RFID is employed on a ubiquitous scale. In contrast to data privacy threats, no personal data is obtained except for the location.

Tracking based on RFID tags provides no continuous data but is much more precise than the tracking of other radio based devices because reader ranges are small compared to for example base stations of cellular radio networks. Furthermore RFID tracking provides additional data such as the concrete type of interaction with an RFID infrastructure.

## 5.3 Current Approaches to Guaranteeing RFID Privacy

So far most RFID privacy mechanisms exist only on paper. This is mainly because privacy is often a secondary issue after security and is often partially satisfied by security mechanisms. Still there exist plenty of suggestions of which an overview is given here. Privacy mechanisms can be divided into tap-proof anti-collision protocols, anonymization of tags, disabling of access and permanent deactivation.

### 5.3.1 Tap-Proof Anti-collision Protocols

When tree-walking anti-collision protocols are used in an RFID system, the unique ID can be eavesdropped by only listening to reader signals [WSRE03]. These signals generally have a longer range than the responses of the tags. The following proposals prevent the determination of the identifier by eavesdropping the forward (reader to tag) channel. The identifier can still be determined by intercepting the backward channel. All of the proposals are based on the assumption that the forward channel has a longer range than the backward channel. This fact, is critically discussed by Thomas Finke and Harald Keller in [FK04].

#### Silent Tree Walking

Silent tree walking is a modification of the anti-collision algorithm suggested by Weis et al. [Wei03]. Instead of calling the next branch in the tree in plaintext, the reader only asks the tags to send the next bit of their unique identifiers. As long as there is no collision all tags share a common prefix. When a collision occurs the prefix is used to conceal the unique part of the identifier. It should be noted that the prefix was sent over the backward channel which is assumed to be harder to eavesdrop on. If  $b1$  is the common prefix and a collision occurred at bit  $b2$ , the reader may either send  $b1 \oplus b2$  or  $b1 \oplus \overline{b2}$ . The tags carry out the same operation and compare the resulting bit with the corresponding part of their identifiers. If there is a match they are selected and send their next bit. An eavesdropper never learns the whole identifiers. Those parts of the identifiers where no collisions occur stay secret.

In contrast to the basic tree-walking algorithm silent tree-walking cannot be implemented on read-only tags because dynamic memory is required.

#### Aloha with Temporary IDs

As an alternative to tree-walking, the specification of the Auto-ID center for class-0 tags [Cen03] provides an anti-collision procedure where the identifiers are never sent over the forward channel (downlink). The tags identify themselves with a temporary identifier which is a random number that is recomputed at every read cycle. These random numbers are used by the reader to individually silence the tags. After all tags have been recognized the actual identifiers are requested via the random numbers. An attacker eavesdropping

on the forward channel only obtains the random numbers that are used for temporary identification. For this anti-collision procedure the tags have to implement a random number generator and the functionality to mute tags.

### 5.3.2 Anonymization of Tags

By anonymization the true identity of the tags is masked. Several protocols for anonymization were assessed for security criteria in [Avo05]. Disappointingly most protocols did not even fulfill the minimum criterium (existential-UNT-QSE). Those who did, suffer from computational complexity.

#### Randomized Hash-Lock

Weis et al. suggest in [WSRE03] a procedure based on the dynamic generation of a new meta-id at every read cycle. On activation, the tags generate a random number and compute its keyed hash based on the identifier. The random number along with the hash is transmitted back to reader. The application now has to compute the keyed hashes of the random number with all identifiers known to it and compare it with the received hash. When there is a match the correct identifier has been found. It should be noted that the reader has to know all the identifiers in its domain. Therefore the mechanism is hardly applicable when a great number of tags has to be dealt with. However the proposal is still interesting for RFID applications because it can be implemented at relatively low costs. Tags need to implement a random number generator.

#### Chained Hashes

The protocol of Ohkubo et al. [OSK03] is very similar to the randomized hash-lock protocol. It also consists of modifying the information sent by the tag each time it is queried by a reader. The difference is that the hash operation is not randomized but is applied on the non-static identifier. For this, the tag needs two hash functions  $h_1$  and  $h_2$ . The current tags identifier must be stored in the database. When a reader queries a tag, it sends  $h_1(\text{ID})$  and replaces its identifier by  $h_2(\text{ID})$ . When the reader receives the tag's response, it forwards it to the database which has to identify the corresponding tag. To do this, the database constructs  $n$  hash chains ( $n$  is the number of tags managed by the database) from the initial identifiers it stores until it finds the expected  $h_1(\text{ID})$ .

#### Henrici and Müller

In the protocol of Henrici et al. [HM04] the tag needs to store a (non-static) identifier ID and two transaction numbers. It guarantees authentication of tags and readers, encrypted communication and assurance of location privacy. No keys or other security relevant data is stored on the tags making attacks on the hardware useless. To guarantee location privacy the ID is altered regularly. The tag never reveals its identifier but only sends hashes. These are calculated based on the transaction numbers, which are synchronized

with the database. This prevents replay attacks and allows the detection of data loss. The drawback of this protocol is that the data management in the backend database is more complex. Although the system is suitable for large-scale applications, it was shown not to be existential-UNT-Q and universal-UNT-QE in [Avo05].

### 5.3.3 Disabling Access

By default RFID tags can be activated by anyone equipped with the correct hardware and without notification of the owner.

#### Blocker Tag

The "blocker tag" proposed by Juels et al. in [JRS03] exploits the tree-walking protocol the reader uses to determine the tags in its field. When the reader queries for tags it can only tell if there's none, one or several tags in its field. In the last case the reader receives overlapping signal responses which cannot be parsed at a time. Instead the reader goes through a tree walking procedure separating the tags' ID space starting with the most significant bit. As the reader walks down the tree, the blocker tag always gives a response regardless of the queried IDs. This way it essentially jams the reader, forcing the reader to unsuccessfully chase down the entire tree. The blocker tag can act either reflexively or transitively. A reflexive blocker tag prevents itself from being read and is similar to the concept of the "kill switch". The main purpose of a transitive blocker tag is to prevent a reader from reading nearby tags. A blocker tag can be customized when to act in as a jammer. For example a blocker tag can be configured to only forbid access to certain areas of the tree.

The Blocker tag can therefore be viewed as an extension of the "kill" switch, with much more flexibility as to whether it responds, and the ability to "kill by association" nearby tags. It does have some problematic aspects, however:

1. While the blocker tag can control whether it responds, again its response is "all or nothing".
2. The blocker tag's response is the same for all readers.
3. The jamming can be overcome. It is possible to determine the presence of a blocker tag. When only one tag responds to a certain bit string as well as to the two variations of the same string with one bit added, the interrogator deals with a blocker tag. Therefore many parts of the tree will be quickly pruned.

### 5.3.4 Permanent Deactivation

A permanent deactivation at a certain point in the use process is the most reliable way to guarantee data and location privacy. However it also prevents the advantages of reuse for example for maintenance or recycling.

## **Kill Command**

As standards are still in their development, the predominantly proposed privacy mechanism is the so-called "kill switch". Each tag has a password and if a kill command is received along with the password, the tag sets an internal bit permanently. The tag no longer responds to interrogations from readers. The typical scenario is that a tag responds to all queries until it is deactivated. Deactivation is usually done after purchase when the customer is exposed to privacy threats. This approach requires only very minimal changes to tag hardware and communication protocols. The proposal is simple and effective, but has some weaknesses:

1. It is an "all or nothing" privacy mechanism - the tag responds to everyone until the kill switch is set, and then responds to no-one. There is no way to have finer-grained disclosure.
2. The user has no way to know whether the tag has actually received the kill command, let alone that the command was interpreted successfully.
3. It appears that the tag will reveal its password to anyone who asks. Therefore it is very easy for malicious readers to kill tags prematurely.

There is also a proposal adding a "conceal" command. As long as the conceal bit is set the tag will still respond to all queries, but with a random ID. The idea is that this allows a count metric without revealing detailed information as to the nature of the items. While this is a positive step away from the "all or nothing" paradigm, it still has the same underlying weaknesses of the kill command.

## **Electromagnetic Deactivation**

Electromagnetic deactivation as it is used in electronic article surveillance can also be used but is not offered up to date. The disadvantages of this approach is that everyone with the dedicated hardware may carry out a deactivation.

## **5.4 Privacy Threats Evaluated**

Currently there exist different opinions on the threat that RFID poses to privacy. For some experts there is none at all, mainly because the amount of data traces left by other devices (credit cards, cell phones) prevails and that the accumulated data has turned out to be hardly used so far. Other experts regard RFID as a very specific threat to privacy with an emphasis on the danger of tracking. There is no controversy on the benefits that come with RFID particularly supply chain transparency.

### **Eavesdropping**

Eavesdropping is an attack that threatens the active as well as the passive party. Possible countermeasures are:



1. Moving data to the backend, and reference only by an identifier.
2. Shielding to prevent a potential attacker from receiving signals.
3. Encrypting the data transmitted over the air interface.

Ideally countermeasures allow the passive party access to data linked to them. However most current practices like storing only identifiers and encrypting data reduce transparency of the system for the passive party.

### **Unauthorized Access**

Unauthorized reading is an attack against the active as well as the passive party. Possible countermeasures are:

1. Detectors that can recognize the supply field of unregistered readers.
2. Authentication mechanisms that allow access for the passive party to guarantee privacy.

Not only reading poses a risk to privacy but also the unauthorized modification of data. This can be prevented by authentication mechanisms. It might be desirable for the passive party to have access to the data on the tag in order to check the correctness of data.

### **Tracking**

Due to the difficulties of installing stealth readers outside legal RFID infrastructures tracking is currently considered rather improbable. Furthermore data that is nowadays being accumulated by non-RFID technologies has so far not been used for tracking and no plans of companies to use data in such a way exist. The primary target area remains supply chain tracking. Auto-checkout and similar applications are not expected to be employed on a large scale before 2010 [fSidI04], mainly because of technical difficulties and costs that are still high compared to other technologies. Apart from that, most companies restrain from putting their customers' trust at stake.

The fragmented nature of the data obtained from readers makes tracking a very difficult task. The efforts for generating a complete movement profile would be enormous and hardly profitable. Even the data recorded from current tracking systems (account cards) often ends up in the data cemetery because the costs of creating profiles are too high.

The situation changes when RFID is employed on a ubiquitous scale because the amount of data traces increases and fine grained tracking becomes possible. Furthermore new parties (police, government) might get an interest on tracking when costs decrease even though tracking with stealth readers will still be improbable. For example if tags are not deactivated after purchase it is possible to determine purchaser and time of purchase from the waste.

## 5.5 Summary

Privacy is a threat that primarily concerns the passive party. The passive party uses tags or objects that are identified by tags but has no control over the data stored on the tags. Privacy can be violated by the active party or a third party.

If an RFID system stores personal data the privacy of the passive party is threatened by eavesdropping or unauthorized access. This is called a threat to data privacy. Location privacy on the other hand is the protection from tracking of tags and reader interactions.

There exist several suggestions to guarantee privacy in RFID systems. Table 5.1 lists the most important generalized approaches.

Threats	Countermeasures
Eavesdropping on the air interface	Moving data to the backend with access for the passive party Shielding Encryption with authorized access for the passive party Attacks for self-protection: Removal of tags Destruction of tags Blocker-tag Jammer Shielding
Unauthorized Access	Detectors Authentication with access for the passive party Attacks for self-protection: Removal of tags Destruction of tags Blocker-tag Jammer Shielding
Tracking	Variable identifiers Attacks for self-protection: Removal of tags Destruction of tags Blocker-tag Jammer Shielding
Non-compliant evaluation of data	No technical countermeasures

Table 5.1: Threats to RFID Privacy and Countermeasures

In [fSidI04] several experts were asked for their opinion on security and privacy threats

for RFID systems. The following list summarizes the main points:

1. Currently the threat of attacks on RFID systems is relatively low.
2. With the increasing use of RFID the threat to security and privacy will rise.
3. RFID security is particularly important when physical security is affected also.
4. Attacks on RFID systems threaten privacy less than common procedures.
5. There are differing opinions on additional privacy risks posed by RFID ranging from zero to very high.
6. Security mechanisms increase fixed as well as variable costs.

Studies carried out in the last few years have shown that consumers are more likely to accept inroads into their privacy under the following circumstances: First they need to be clearly aware of the benefits of the new technology, secondly people feel more secure if there is a human they can contact in case of problems. Thirdly people are less unsettled if the new technology is deployed by a renowned company. Even though privacy concerns exist they are not likely to prevent a large-scale breakthrough of RFID alone.

## Chapter 6

# RFID Applications

Since RFID is a new technology, there exists a lack of standards and missing experience in the field. However there are large retail industry pilot projects currently carried out and studied. Nevertheless, only few studies analyze the actual cost savings and benefits of RFID deployment. This chapter presents the possible application areas and specific implementations of RFID. Finally an analysis of the chances and risks of RFID is performed.

### 6.1 Application Areas of RFID

Even though RFID is a technology that can be applied in numerous business scenarios its prime functionality is the identification of objects. With this background the following sections describe the most important application areas of RFID.

#### 6.1.1 Access and Route Control

Convenient RFID tags can replace magnetic cards and chip cards for access control or bank accounts. Users only have to position their card near a reader. Consequently high standards for security mechanisms are required in order to avoid fraud. For most applications the familiar plastic card format is used but housing the chip in watches or key fobs is also an option. Typical frequencies are in the 13,56 MHz area. Readers support ranges of up to one meter. Apart from identification writing and updating tags is also possible. Access control systems can increase efficiency whenever a great number of people have to pass the same access point as it is the case in large companies and holiday resorts.

Recently greater airlines have started using RFID for baggage routing to reduce errors. Another popular application area of RFID is the packet routing within companies. Usually only simple tags with a unique identifier are required. Tracking information is stored in a central database, which is updated whenever the packet passes a control point. This way customers can query the state of their order via internet.

### 6.1.2 Document Verification

Current pilot projects deal with the use of RFID in identity cards and travel documents. The tag is used to implement anti-forgery mechanisms and in further consequence provide extended verification capabilities. These mechanisms include the saving of biometric data, such as face and fingerprints, on the tag. There is a tendency towards cross-linking different identification features thus creating a multi-biometry platform to compensate for weaknesses of individual technical methods.

### 6.1.3 Asset Management

Particularly airports and vehicle factories, where the asset management comes in as a major cost factor, can benefit from RFID. Firstly shrinkage and theft can be eliminated, secondly many processes can be optimized. In [LSF04] Lampe Strasser and Fleisch suggest a ubiquitous computing environment based on RFID for maintenance, repair, and overhaul (MRO) of aircrafts where strict regulations define requirements for quality, safety, and documentation.

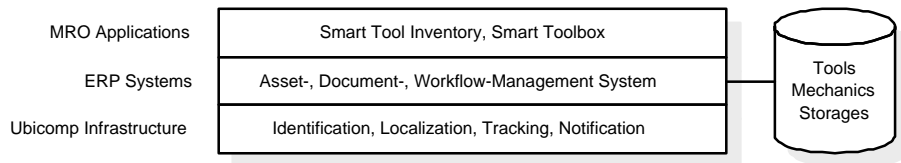


Figure 6.1: Asset Management Based on RFID

MRO costs are corresponding to 12 % of the total operating costs of an aircraft. During maintenance of commercially used aircrafts the owner faces high opportunity costs. Therefore RFID brings significant competitive advantage by:

1. *Avoidance of delays.* Search actions for parts, tools, or documents are eliminated.
2. *Avoidance of human errors.* The right parts and tools can be identified, and no tools get misplaced or lost. This results in higher quality and security.
3. *Automation of documentation.* Actions, tool use, and completeness checks are documented automatically.
4. *Efficient use of resources.* The use of mechanics, parts, and tools is planned and monitored.

### 6.1.4 Supply Chain

The supply chain is a multi-stage process, which involves everything from the supplying of prime materials, used to develop products, to the products delivery to customers via warehouses and distribution centers. Supply chains exist in both service, manufacturing and retail organizations. Although, the complexity of the chain changes greatly from one industry branch to another, its management can be seen as the organization of the flows of these materials, as they move through the various processes. The efficiency of the supply chain has a direct impact on the profitability of a company. Therefore any major company striving for competitive edge needs to invest in infrastructures to control inventory, track products and manage associated finance.

By increasing transparency in the supply chain, RFID allows the optimization of logistic processes. The primary goal is the discovery of inefficiencies in the value chain within and between the companies thus rationalizing the material, information and financial flows. RFID enables the fine grained tracking of lot sizes down to one, over the entire logistic network, thus facilitating the detection and the locating of losses and shrinkage, the result of misplaced orders, theft and inefficient stock management.

## 6.2 Implementations

Now that the general application areas of RFID have been presented, this section discusses the most important systems and recent developments that are based on RFID.

### 6.2.1 The EPCglobal Network

The Auto-ID center at the Massachusetts Institute of Technology (MIT) along with several business and academic leaders have designed a product portfolio to introduce RFID to the global supply chain. It consists of the Electronic Product Code (EPC), RFID technology and supporting software based on the EPC standard and is referred to as the EPCglobal Network. The EPCglobal Network will be able to provide real time data about individual items as they move through the supply chain. It consists of 5 components:

1. *Electronic Product Code:* Unique number that designates the item in the supply chain.
2. *ID System:* The ID system consists of EPC tags, which are applied to the items and EPC readers, which can read out the EPC stored on the tags via RFID. EPC readers communicate with the local business information systems using EPC Middleware.
3. *EPC Middleware:* The EPC Middleware manages the read data (filtering sorting) and passes on relevant data. Timestamp and location are added so that the EPC Information Services and local information systems can work with the data.

4. *Discovery Services*: Discovery Services are a suite of services that allows users to find data associated with a specific EPC and to request access to the data. The Object Naming System (ONS) is included here.
5. *EPC Information Services (EPCIS)*: This service allows users to exchange information with trading partners through the EPCglobal Network.

In a typical application EPC tags are affixed to items and readers are placed at strategic positions such as gates. Readers track time location and EPC. The tags and readers are integrated with the local infrastructure at the individual sites. Once the information is captured the EPCglobal Network uses internet technology to share the information with authorized trading partners as illustrated in figure 6.2. Those partners use Discovery Services to obtain data about a specific EPC. The actual data is obtained from the EPC Information Services from local servers that store information about the EPC. No information except for the EPC is stored on the tag. The result is a network that traces products in the supply chain in real time.

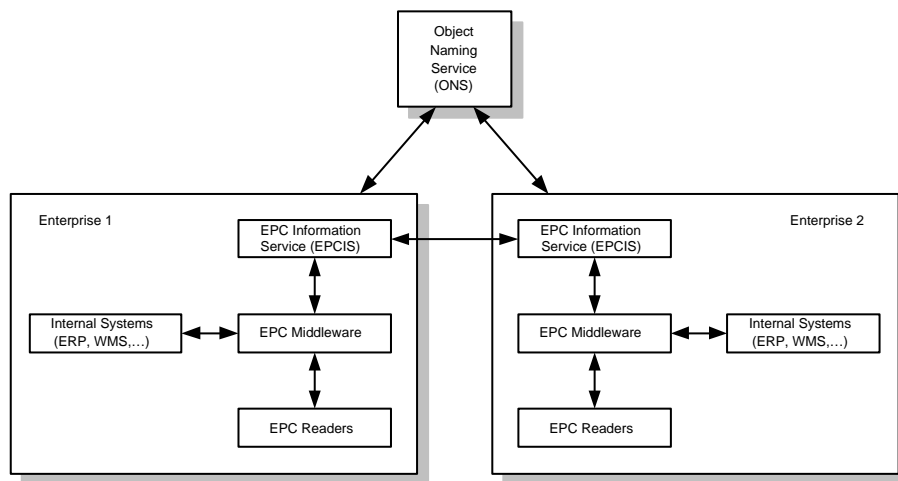


Figure 6.2: The EPCglobal Network

There are ongoing discussions on the harmonization of EPC and ISO18000 air interfaces for the UHF frequency band. It should be noted that EPCglobal specifies a complete RFID system including the application layer, which is not the case with ISO18000.

The EPC is a standardized and unique number in the EPCglobal Network. No information beyond the number is included in the EPC. All information related to a specific EPC can be found in the EPCglobal Network and is protected by firewalls, encryption and other security measures [Gra04].

### 6.2.2 E-Government

E-Government refers to the use by government agencies of information technologies (such as Wide Area Networks, the Internet, and mobile computing) that have the ability to transform relations with citizens, businesses, and other arms of government. This includes the issuing of smart cards to citizens. Austria's E-Government system supports the following functions:

1. Generating and checking of digital signatures.
2. Encryption and decryption of electronic documents.
3. Calculating and checking of hash values over documents.
4. Storing and reading of government specific data.

These functionality is provided by the Citizen Card Environment. The citizen card itself only stores basic personal information (first name, last name, a unique id and keys). The citizen card does not necessarily have to be a smart card, which requires the user to purchase a reader. USB tokens or NFC devices may be used as well. The citizen card concept acts as national standard so that users may choose from several vendors.

Private keys for signature creation are stored on the cryptographic card and consequently can't be accessed. The public keys are made publicly available, so that everybody can verify the signatures. Private keys are certified by the certification authority binding the identity of the card owner to his signature.

Austrian law stipulates the use SHA-1 to compute the hash value and of RSA for encrypting the hash value [Öst04].

### 6.2.3 Near Field Communication (NFC)

NFC evolved from the RFID technology and is designed for interactions between tags and electronic devices in close proximity (<10 cm). In 2005 first devices supporting the Near Field Communication Protocol will be delivered. NFC is compatible with Sony's FeliCa chip, which is used in Japan for security relevant applications such as e-wallets, ticketing and locks.

The Near Field Communication Interface and Protocol specifies the communication mode selection mechanism. This protocol deals with the situation that the NFC protocol, ISO 14443 and ISO 15693 devices all operate at 13,56 MHz, but with different protocols. It is specified that NFC compliant devices can enter each of these three communication modes and are designed not to disturb other RF fields at 13,56 MHz.

Since the protocol's target application includes consumer electronics the system was designed to provide ease of use. NFC works at ranges of a few centimeters. To establish a connection two devices simply have to be put in close range to each other. A peer to peer connection is created automatically and data can be exchanged. NFC can also be used to configure faster and longer range protocols like Bluetooth or Wireless Ethernet WiFi.



NFC works in the unregulated 13.65 MHz band. Operating distances of 0-20cm are under consideration. Communication is half duplex and based on the listen before talk principle, meaning devices listen if the carrier is currently in use before they start transmitting. Any device can be either initiator or target. The initiator initiates and controls data exchange while the target listens and answers requests. There are two modes of operation:

1. In *active mode* both devices generate their own RF field to carry the data.
2. In *passive mode* only the initiator device generates the RF field while the other uses load modulation to transmit the data.

The application sets the initial communication speed to 106, 212 or 424 kbit/s. Different modulation schemes and bit encoding mechanisms are used depending on the current speed. Communication security is guaranteed by the fact that the devices must be in close range (almost touching each other). The passive mode of NFC allows energy-critical applications to save power because only one of the devices needs to generate the RF field. NFC communication is compatible with the FeliCa and the MiFare protocols and can also work with smart cards and smart card readers. An NFC device can view cards and can also act as a card.

Secure NFC combines NFC applications with smart card security. Devices act like contactless smart cards with cryptographic capabilities.

## 6.2.4 Machine Readable Travel Document (MRTD)

Currently under way is the specification of the new European Passport, which stores biometric data (two fingerprints and photograph) in machine readable format using RFID technology. The responsible entity is the International Civil Aviation Organization (ICAO). The data storage is based on the ISO 14443 proximity standard Type A and B. The main reasons for using this standard is that it best meets storage space requirements (64 kB needed for photograph and fingerprints) and data transmission speed. ISO 15693 vicinity supports only 15 kB so far and reading would take unacceptably long.

Security during production is guaranteed by tight security measures and control mechanisms preventing internal fraud. After issuing the passport is locked and cannot be rewritten. In addition to passive authentication by digital signatures, states may choose optional security mechanisms, using more complex ways of securing the chip and its data.

In order to avoid skimming and misuse the MRTD can only be read when it is opened and after the Optical Character Recognition (OCR) number has been transmitted by the reader. Readers therefore have to be capable of reading the machine readable zone (MRZ) and the PICC simultaneously. An overview of the security mechanisms employed for MRTDs see table 6.1. For a detailed description see [Org04c].

In a typical reading process first the digital signatures are checked. If they are valid the MRZ is checked against the chip. If that step succeeds the MRTD is allowed access. If one of the steps fails closer human inspection is required.

## 6.3 RFID - Chances and Risks

The pilot projects that were carried out in several application areas and with various degrees of complexity showed that there exist supportive but also inhibitory factors for the employment of RFID. RFID systems compete with other auto-id systems such as barcodes, OCR, and chip cards. The table 6.2 from [Fin03] compares important system parameters of these technologies.

Compared to other technologies RFID has a clear advantage in capacity, which comprises data quantity, data density and reading speed. Further advantages are the resistance to environmental influences, such as dirt and damp, and the non-existing line-of-sight restriction.

Despite the advantages given, there are many factors that make the decision of deployment of RFID, still risky. The high implementation costs of RFID can be partially relativized in view of the projected cost savings. However the RFID world is still lacking standards and there is an uncertainty if RFID will be fully accepted by the industry and if it can be successfully integrated in current business processes.

Next to these points there are also external factors that should be considered.

### 6.3.1 Economical Aspects

The rising pressure on companies to reduce costs and increase competitive edge on the international markets will clearly support the adoption of RFID. There are significant chances where productivity can be raised by increased automatization. Here the performance that can be gained by a transparent supply chain and reduced transaction costs is particularly high. But also the rising interconnection of markets will support RFID. By a systematic analysis of relationships in the complex logistic network, inefficiencies can be discovered and eliminated. RFID has also the potential to automatically customize machines in the production process, thus making mass customization possible.

### 6.3.2 Legal Regulations

Companies usually face a great number of legal regulations regarding their products. These regulations usually cover retraceability, quality standards, security and documentation, which can be supplied by the increased transparency that comes with the use of RFID. Retraceability is achieved with a detailed logging of stations the product passes. Time logging and the logging of additional parameters such as temperature or pressure can be used to guarantee quality standards. The storage and transportation of dangerous goods are under particularly tight restrictions. These processes can be conveniently documented to the responsible authority via RFID.

### **6.3.3 Technical Aspects**

When deciding on the installation of an RFID system some technical problems that remain unsolved have to be considered. For example there are significant difficulties in the detection of tags near fluids and metal objects in certain frequency ranges. With increasing data quantities on the tag the reading times also increase up to unacceptable lengths. Furthermore there might occur shadowing effects between similar signals, interference as well as frequency shifts. To avoid negative surprises these effects have to be carefully considered and tested, prior to a large scale deployment.

### **6.3.4 Standardization**

Even though the international standardization advances there are still many subareas that are largely unregulated. A worldwide standardization is mandatory for hardware and software producers to rely on technical parameters and for the customers to use products from different producers. A lack of standards leads to inefficiencies because customers have to rely on a single equipment provider. Even the well known EPC standard is not yet fully standardized in its details. Another problem is that frequency regulations are neither internationally standardized. Currently multinational companies have to use tags that support different frequency ranges. Recently Japan allowed the UHF band used in the new EPC Gen2 UHF standard.

### **6.3.5 Integration Costs**

Currently the prices of tags are still too high for many companies to make RFID an interesting investment. However business analysts project that the tag costs will be falling rapidly with increasing mass production. Next to the tag costs significant investments in the infrastructure have to be made. This includes equipment, such as terminals and networks, for the collection, processing and evaluation of the data supplied by the RFID system. Additionally the restructuring of business process and parallel operation during the initial phase are also major cost factors.

### **6.3.6 Security**

RFID systems provide a maximum congruency between the physical and the virtual world. With the widespread deployment of RFID there will be a rising dependency on the correctness of the data supplied. This leads to risks that can only be guaranteed with adequate security mechanisms. Current experiences show that even mechanisms that have so far been considered safe can become insecure with technical progress. Consequently the discovery of security vulnerabilities can lead to a significant capital expenditure due to an infrastructure update.

### 6.3.7 Privacy

The increasing deployment of RFID is warily eyed by the media and the public. From the public point of view the guarantee of privacy and data protection is an important issue. Companies will restrain from upsetting their customers. However most customers will soon be persuaded by the increased convenience and benefits that RFID can provide.

## 6.4 Summary

The predicted main application area where the greatest benefits can be expected is the supply-chain management. This is the use of information technology to give automated intelligence to a network of vendors, suppliers, manufacturers, distributors, retailers, and a host of other trading partners. The goal is for each player in the supply chain to conduct business with the information from others in the chain, guiding supply and demand into a more perfect balance. Effective management of the supply chain enables a company to move product from the point of origin to that of consumption in the least amount of time at the smallest cost.

A great number of reference and pilot projects are currently under way. The following factors are supportive for a large scale breakthrough of RFID technology:

1. Economical aspects.
2. Legal regulations.

On the other side there are factors that are inhibitory for the deployment of RFID:

1. Technical Problems.
2. Lack of Standardization.
3. Integration costs.
4. Security measures.
5. Unresolved privacy issues.
6. Lack of experience.

The reason that a large scale adoption has not yet taken place is that the costs of hardware and software components are still high. The necessary adaptations to business processes are also associated with hidden costs. As of now there exist only few studies documenting the optimizations that can be accomplished with the deployment of RFID. The application of RFID is strongly backed by the retail industry including large companies like the Metro Group, Wal-Mart and Tesco. The success of these pilot projects will play a decisive role in the breakthrough of RFID. Current estimations say that RFID can reduce storage costs by 5% and personnel costs by up to 10% [Kea03]. Economies of scale exist. This is the reason why the European big players in the retail industry are pressuring the use of RFID by their suppliers.

Method	Benefits	Deficiencies
Passive Authentication	Proves that the contents are authentic	Does not prevent an exact copy or chip substitution. Does not prevent unauthorized access. Does not prevent skimming.
Comparison of OCR-B and LDS	Proves that chip content and MRTD belong together.	Does not prevent an exact copy of chip together with document.
Active Authentication	Prevents copying of the security object and proves that security object is read from the authentic chip.	Requires processor chips.
Basic Access Control	Prevents skimming and misuse. Prevents eavesdropping on the communications between MRTD and inspection system (when used to set up encrypted session channel).	Does not prevent an exact copy or chip substitution (requires also copying of the conventional document). Requires processor-chips.
Extended Access Control	Prevents unauthorized access to additional biometrics. Prevents skimming of additional biometrics.	Requires additional key management. Does not prevent an exact copy or chip substitution. Requires processor-chips.
Data Encryption	Secures additional biometrics. Does not require processorchips.	Requires complex decryption key management. Does not prevent an exact copy or chip substitution.

Table 6.1: Security Mechanisms for MRTDs

System parameters	Barcode	OCR	Chipcard	RFID
Typical data quantity (bytes)	1-100	1-100	16-64k	16-64k
Data density	Low	Low	Very High	Very high
Machine readability	Good	Good	Good	Good
Readability by people	Limited	Simple	Impossible	Impossible
Influence of dirt/damp	Very high	Very high	Possible (contacts)	Impossible
Influence of (opt.) covering	Total failure	Total failure	Possible	No effect
Influence of direction and position	Low	Low	Unidirectional	None
Degradation/wear	Limited	Limited	Contacts	None
Purchase cost/reading electronics	Very low	Medium	Low	Medium
Operating costs	Low	Low	Medium	Medium
Unauthorised copying/modification	Easy	Easy	Difficult	Difficult
Reading speed (including handling of data carrier)	Low 4s	Low 3s	Low 4s	Fast 0,5s
Maximum distance between data carrier and reader	0-50 cm	<1 cm Scanner	Direct contact	0-5m

Table 6.2: Comparison of Auto-ID Systems [Fin03]

## Chapter 7

# Securing a Distributed RFID Infrastructure

Most RFID systems contain a great number of RFID devices such as printers and readers, distributed over a site. For the management of these devices and the management of the data supplied by these devices a central point of access is necessary. This necessitates a special client server architecture, which is presented in this chapter. A threat model for this architecture is developed and a security solution is presented.

### 7.1 RFID Infrastructures

In this section we will take a look at the requirements for RFID infrastructures as well as their hardware and software components. Based on this information a typical RFID middleware architecture will be presented.

#### 7.1.1 Requirements

Effective RFID architectures can borrow from the principles developed for financial trading systems, process control and large-scale network management [Pal04]. Like RFID systems, these systems process huge amounts of data, correct errors in real time, correlate events, detect patterns, re-organize and cleanse data and recover from faults ideally in real time. RFID architectures should regard three central principles of these systems.

**Operational data management architecture.** An operational data management architecture is a software system that captures events at the "edge" of the enterprise, where operational activity occurs, rather than in the center, where business-oriented transaction processing occurs. These architectures act as a cache for transactional, enterprise systems and then communicate via middleware once actionable information is identified.

Operational data management architectures are common in the financial industry. For example, trading systems collect market data in real time, execute trading strate-

gies heuristically, identify trading opportunities and forward the results to traders for action. These systems eventually generate the trades tracked by back-office systems, but billions of dollars are invested every year on the operational stages, that precede the actual trade. This kind of data management and applied intelligence will become more prevalent as RFID adoption encourages companies to deploy intelligence at the point of operational activity.

**Concentrators.** One way to deal with the great amounts of RFID data is to develop concentrators that help control the flow and provide filtering and aggregation of EPC event streams. Savants are distributed software systems developed by the Auto-ID Center to act as the central nervous system of the EPC Network. A Savant takes data from an RFID reader, does some filtering, handles product lookups and sends the information on to enterprise applications or databases. Savants provide interfaces to register a user EPC event receiver. A Savant concentrator is a software component that would run on a server and apply intelligence to control the volume of data.

The concentrator is a type of operational architecture that starts with the Savant, which grabs EPC data from readers, stores the data in an operational database, checks for errors and operational anomalies and applies business intelligence to identify critical business-level events, then propagates the data via the enterprises middleware to another database, to a user or to both. This architecture has multiple layers of middleware, including Savants, message-oriented middleware and enterprise application integration, as well as an operational data store and an enterprise data store.

**Pipelines.** As the project scope expands, data volume grows far more rapidly than often expected. Thus the system must be scalable and flexible. Increasing numbers of RFID readers will lead to more streams of data, which can quickly overwhelm the system. One way to handle the load is through pipeline processing.

Once concentrators have been placed, pipelines are set up to handle the streams of data. Pipelines separate streams of EPC data to handle load and coordinate or process the data streams after they have been captured. To achieve additional scalability more than one concentrator may be installed on a set of distributed machines, each one controlling a domain of RFID activity, distributed over multiple machines. For instance, EPC readers at a distribution center with 30 dock doors may be needed. Half the readers are tied into one concentrator and half into another. One pipeline will feed data into a local inventory database, one into a regional database and one into financial applications.

By following these principles RFID architecture will achieve a maximum of scalability and flexibility.



## 7.1.2 Architecture

An RFID system consists of tags, printers, readers and middleware. Tags include a chip and an antenna embedded in a label. RFID printers simultaneously print labels and write to the electronic tag. Readers, which can be stationary or mobile, collect data from the tag and may apply a first level of intelligence, filtering out duplications and noise. Middleware turns the raw data into useable information and transmits it to back-end systems. Tags are a large and recurring expense, which is why there is so much concern over their price. There is also a concern over yield.

### Tags and Printers

Most supply chain pilot programs use rolls of adhesive-backed paper shipping labels that are inlaid with a plastic-enclosed chip and antenna. The labels are printed and the tags encoded by an RFID-enabled printer in the warehouse before being attached to a case or pallet. Converting RFID tags into printable labels is a multi-step process that has required a learning curve. Intelligent printers can perform testing when the inlay is put into the label. Tag placement is also an issue in determining how well an RFID system performs. RFID is constrained by basic physics of the radio wave. With metal and liquid and longer distances, systems may encounter problems.

### Readers

Tags operate in conjunction with readers that collect the raw encoded data and need to be compatible with the type of tags in use. Portal readers are usually positioned over conveyers and at dock doors. For stationary readers some engineering expertise is required to ensure that they are positioned correctly. Mobile readers are in most cases hand-held devices similar to barcode readers but may also be integrated into forklifts or other vehicles.

### Middleware

The bulk of the return on investment for RFID tagging will come from intelligent use of the generated data. The RFID middleware manages the flow of data between tag readers and enterprise applications and is responsible for the quality and usability of the information. The elements of the middleware include the following:

**Reader and device management.** RFID middleware should allow users to configure, monitor, deploy and issue commands directly to readers through a common interface.

**Data management.** As RFID middleware captures EPC data or other data from readers it should be capable of intelligently filtering and routing them to their appropriate destinations. This capability should include both low-level logic like filtering out

duplicate reads and more complex algorithms like content-based routing. Comprehensive solutions will also offer tools for aggregating and managing EPC data in either a federated or central data source.

**Application integration.** RFID middleware solutions should provide the messaging, routing and connectivity features required to reliably integrate RFID data into existing SCM, ERP, WMS, or CRM systems, ideally through a service-oriented architecture. It should also provide a library of adapters to popular WMS and SCM applications like SAP or Manhattan Associates, as well as APIs and adapters for using standard technologies like JMS, XML and SOAP to integrate with other third-party applications.

**Partner integration.** Some of the most promising benefits of RFID will come from sharing RFID data with partners to improve collaborative processes like demand forecasting and vendor-managed inventory. This means that RFID middleware must provide B2B integration features like partner profile management, support for B2B transport protocols and integration with the EPCglobal Network.

**Process management and application development.** Instead of just routing RFID data to business applications, sophisticated RFID middleware platforms will actually manage RFID related processes that touch multiple applications and enterprises, like inventory replenishment, key process management and composite application development features include workflow and role management and process automation.

**Packaged RFID content.** RFID middleware platforms that include packaged routing logic, product data schemas and integration with RFID-related applications and processes like shipping, receiving and asset tracking are major assets. This content gives firms an easy start on their RFID projects.

**Architecture scalability and administration.** It can be said with certainty that RFID adoption is going to produce a lot of data and the RFID middleware is the core for reliably processing that data. This means that RFID middleware platforms must include features for dynamically balancing processing loads across multiple servers and automatically rerouting data upon server failure. These features should span all tiers of the architecture even the edge devices.

**Performance.** Since the data loads processed by an RFID can be very high, performance is another key requirement. This can be achieved by dynamic load processing, filtering of data as well as the preprocessing of data.

**Security.** The middleware usually offers a central point of access to the data processed. Furthermore a configuration entity in the middleware allows the management of devices. As a consequence it is absolutely necessary to ensure that the system meets all the security requirements defined.

Summarizing, the core functionality of RFID middleware are data and device monitoring and management. It extracts data from RFID readers, filters and aggregates information and routes it to enterprise systems (see image 7.1 from [Lia04]).

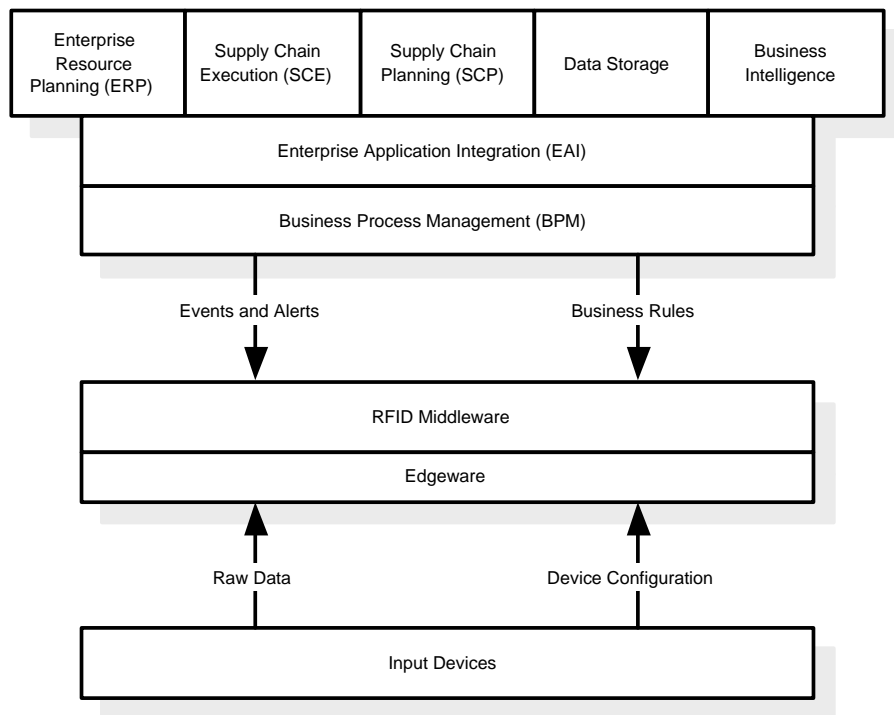


Figure 7.1: RFID Middleware

### Enterprise Applications

The middleware links data obtained from the RFID hardware to enterprise applications, where the data are processed. Based on the data gathered decisions can be automated or be supported by information systems.

**Warehouse Management Systems (WMS)** integrate mechanical and human activities with an information system to effectively manage warehouse business processes and direct warehouse activities. These systems automate receiving, put-away, picking and shipping in warehouses and can prompt workers to do inventory cycle counts. Most support radio-frequency communications, allowing real-time data transfer between the system and warehouse personnel.

**Materials Requirements Planning (MRP and MRP II)** are phases in the development of computerized methods for planning the use of company resources, includ-

ing scheduling raw materials, vendors, production equipment and processes.

**Enterprise Resource Planning (ERP)** is the current evolution of manufacturing resources planning (MRP and MRP II) systems. ERP is being positioned as the foundation and integration of enterprise-wide information systems. Such systems will link together all of a company's operations including human resources, financials, manufacturing and distribution as well as connect the organization to its customers and suppliers.

**Manufacturing Execution Systems (MES)** are systems that use network computing to automate production control and process automation. By downloading manufacturing plans and work schedules and uploading production results, MESs bridge the gap between business and plant-floor or process-control systems.

**Customer Relation Ship Management (CRM)** focusses on service automated processes, personal information gathering and processing and self-service. It attempts to integrate and automate the various customer serving processes within a company. RFID enriches CRM in that it can automatically track customers on one hand and in that it can provide customers with information on the supply chain of a company's products on the other hand.

## 7.2 Threat Model

This section will create a threat model for typical RFID applications. In order to do this, first the assets to be protected will be identified. Subsequently a model for an RFID application is created and analyzed for security.

### 7.2.1 Identification of Assets to Protect

The security solution for our RFID infrastructure must guarantee that the correct functionality of the system is maintained. There are two kinds of data in the system which must be protected: EPC data and configuration data.

The main source of EPC data is the reader interface from where the data are generated and passed on for further processing. Here the data must be guaranteed to be authentic and complete. Furthermore a breakdown results in data loss which leads to significant costs considering the huge amounts of data usually processed by an RFID system.

The same rules apply for configuration data which are exchanged between a configuration tool and the devices, except that a violation of availability does not have such severe effects as is the case with EPC data.

Apart from the data that must be protected there are three entities in an RFID infrastructure. First of all there are the RFID devices such as readers, printers or light barriers. Secondly there must be an entity allowing the central configuration of devices and business logic. And thirdly there must be an entity which processes the data supplied by the devices, applies a logic to them and provides an interface to enterprise applications.

For the RFID devices it is very important to guarantee availability and generate warnings if a violation of it occurs. To prevent the installation of rogue readers, the readers are ideally authenticated to the system.

The readers are configured and managed from a central point, the administration entity. Therefore this entity has significant importance and must be protected by authentication. For the information entity authentication is optional because it only provides the data collected.

### 7.2.2 System Model

Based on the requirements presented in the previous section the security research is based on an architecture as presented in figure 7.2. Security on the air interface was already discussed in the previous chapters. The distributed RFID infrastructure model is made up of three components.

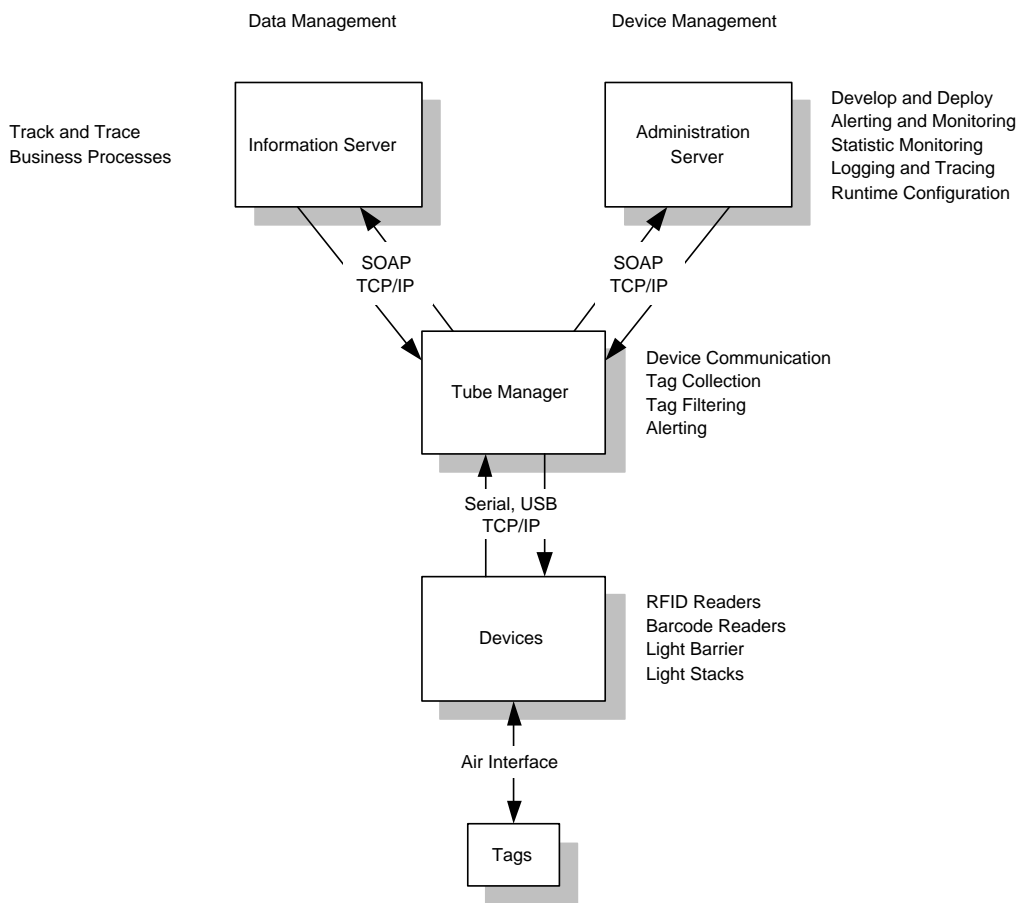


Figure 7.2: RFID Middleware Architecture

The *Tube Manager* is a platform independent application that acts as a device controller. It interfaces a variable number of RFID devices allowing for a scalable architecture. These RFID devices can be a combination of readers, RFID printers, barcode readers and other RFID-related devices. The tube manager hosts all tube specific programs and configurations, which can be dynamically downloaded and updated during operation. In order to minimize network traffic, rules can be defined to build packet transmission. It also has built-in health state information about devices. Alerts are configurable.

With the *Administration Server* the RFID devices and Tube Managers in the RFID infrastructure can be managed. In order to do this, it uses the Tube Manager as an interface to the devices. The Tube Managers forward their status information to the Administration Server, where the data are processed and logged. The Administration Server further lists status information and other device related data such as reader firmware versions and installed components. Apart from displaying information about the readers, the Administration Server is also used to configure and manage the settings of the devices in the infrastructure. When devices are deployed, their configuration is performed from within the Administration Server.

In contrast to the Administration Server that handles the configuration of devices, the data is handled by the *Information Server*. This includes a complete data log from the tube managers to a database. A relation to location, time and tube is stored along with the events. The information server has a link to the administration server in order to get access to tube and device configuration information. Its track and trace module allows the following of tags in the system across different tubes and devices and enables linking tags to items. Users can set filters and generate reports automatically. The EPC commissioning tool allows the assigning of EPC numbers from a global repository and other EPC related functionality.

### 7.2.3 Identification of Entry Points

With the architecture defined it is now possible to identify possible entry points for an attacker. Depending on the point where the attack takes place the danger emanating from the attacks varies. Particularly dangerous are attacks targeted at the administration and the information server because they act as a central control point for device and data management. In the following paragraphs the possible attacks at the different entry points are analyzed and qualified. Possible entry points are depicted in figure 7.3.

**Unauthorized access** can occur at every entity in the data flow chain, but is significantly important for the client if it is located outside the local network. But also in the local network an entity might get unauthorized access to the administration server or to the tube manager directly. A client gaining unauthorized access to the administration server can tamper with the configuration of the devices in the RFID system and gather information concerning their configuration. He has also the ability to install rogue devices and can change the business logic. The great danger in this case is, that the attacker has access to the central point of configuration for all devices. In contrast to that, an attacker

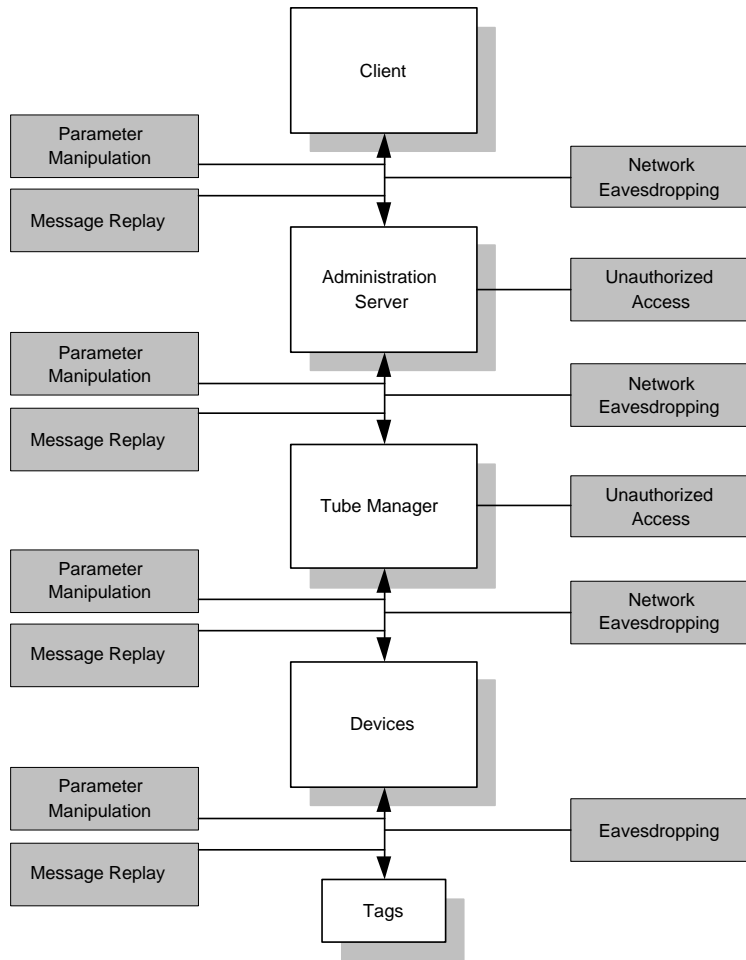


Figure 7.3: Possible Entry Points

in the local network who gains access to a tube manager may only change the devices attached to it. The same applies for the information server. An attacker gaining access to it, has an overview over all the collected data in the system plus their associations, while access to a tube manager only provides data from the devices in the tube, which is generally useless without their associated entities. Vulnerabilities that can lead to unauthorized access are:

- Missing authentication.
- Passwords passed in plaintext in SOAP headers.
- Basic authentication over an unencrypted channel.

Countermeasures include:

- Password digests in SOAP headers for authentication.
- Kerberos tickets in SOAP headers for authentication.
- X.509 certificates in SOAP headers for authentication.

**Parameter manipulation** refers to the unauthorized modification of data sent between the web service consumer and the web service. An attacker might, for example, intercept a web service message and modify it before sending it on to its intended endpoint. Parameter manipulation is therefore an attack on the integrity of data. As with unauthorized access an attack of this kind is particularly harmful when directed against the administration or the information server. Modified messages between the client and the administration server can trigger unwanted configuration events or be used to reroute useful information. For a bidirectional message modification between the administration server and the tube manager can lead to misconfiguration of devices attached to the tube or generate false status messages which can possibly lead to a violation of availability. Between the information server and the tube manager parameter configuration can lead to invalid data, which might be automatically discovered by the business logic configuration. Vulnerabilities that can lead to parameter manipulation are:

- Messages that are not digitally signed to provide tamperproofing.
- Messages that are not encrypted to provide privacy and tamperproofing.

Countermeasures include:

- Digitally signing the message.
- Encrypting the message payload to provide privacy and tamperproofing.

With **network eavesdropping**, an attacker is able to view web service messages as they flow across the network. For example, an attacker can use network monitoring software to retrieve sensitive data contained in a SOAP message. This might include sensitive application level data or credential information. As long as the data are not modified, the danger is limited if not used as a basis for further attack. In this case the configuration data need to be paid special attention to. The information server might be an attack point for espionage reasons depending on the importance of the data. The path from the servers to the tube managers is no critical because information is out of frame. Vulnerabilities leading to network eavesdropping are:

- Credentials passed in plaintext in SOAP headers.
- No message level encryption used.
- No transport level encryption used.

To avoid the problem of network eavesdropping the following methods can be employed:



- Transport level encryption such as SSL or IPSec can be used. This is applicable only if both endpoints are controlled.
- Alternatively the message payload may be encrypted to provide privacy. This approach works in scenarios where messages travel through intermediary nodes to the final destination.

Web service messages can potentially travel through multiple intermediate servers. With a **message replay** attack, an attacker captures and copies a message and replays it to the web service impersonating the client. The message may or may not be modified. In a basic replay attack the attacker captures and copies a message, and then replays the same message and impersonates the client. A replay attack does not require the malicious user to know the contents of the message. In a man-in-the-middle attack, the attacker captures the message and then changes some of its contents, for example, a shipping address and then replays it to the web service. When the message is modified, the dangers are similar to those with parameter manipulation. If not the danger of these attacks is limited. Message replay is possible when:

- Messages are not encrypted.
- Messages are not digitally signed to prevent tampering and
- duplicate messages are not detected because no unique message ID is used.

To prevent message replay:

- An encrypted communication channel such as SSL may be used.
- The message payload may be encrypted to provide message privacy and tamper-proofing. Although this does not prevent basic replay attacks, it does prevent man-in-the-middle attacks where the message contents are modified before being replayed.
- The use of a unique message ID or nonce with each request can be used to detect duplicates and digital signatures to provide tamperproofing.

## 7.3 Implementing Security

Now that a threat model has been created various security mechanisms are suggested, compared and assessed. First the layers on which security can be implemented are presented. Then the mechanisms are discussed in more detail.

### 7.3.1 Security Models

The **transport-level security model** is simple, well understood and adequate for many scenarios, in which the transport mechanisms and endpoint configurations can be tightly controlled.

The main issues with transport-level security are that security becomes tightly coupled to and dependant upon the underlying platform, transport mechanism and security service provider. Security is applied on a point to point basis, with no provision for multiple hops and routing through intermediate application nodes.

For **message level security**, the WS-Security specifications describe enhancements to SOAP messaging that provide message integrity, message confidentiality and single message authentication. Authentication is provided by security tokens, which flow in SOAP headers. No specific type of token is required by WS-Security. The security tokens may include Kerberos tickets, X.509 certificates, or a custom binary token. Secure communication is provided by digital signatures to ensure message integrity and XML encryption for message confidentiality.

WS-Security can be used to construct a framework for exchanging secure messages in a heterogeneous web services environment. It is ideally suited to heterogeneous environments and scenarios where endpoints and intermediate application nodes are not under direct control. Message-level security:

- Can be independent from the underlying transport.
- Enables a heterogeneous security architecture.
- Provides end-to-end security and accommodates message routing through intermediate application nodes.
- Supports multiple encryption technologies.
- Supports non-repudiation.

### **7.3.2 Authentication and Authorization**

Since the administration server and the information server output sensitive, restricted data and provide restricted services, they need to authenticate callers. Secondly a meaningful authorization policy requires authenticated users.

#### **Authentication Schemes**

If a web service outputs sensitive, restricted data or if it provides restricted services, it needs to authenticate callers. A number of authentication schemes are available, which can be broadly divided into three categories:

- Platform level authentication,
- message level authentication and
- application level authentication.

If both endpoints are under control and both endpoints are in the same or trusting domains, **platform authentication** can be used to authenticate callers. With basic authentication the user must configure the proxy and provide credentials in the form of a user name and password. The credentials are transmitted in plaintext and therefore basic authentication should only be used along with SSL. In Windows systems the web service's virtual directory can be configured for integrated platform authentication, which results either in Kerberos or NTLM authentication to be used, depending on the client and server environment. The advantage of this approach in comparison to basic authentication is that credentials are not sent over the network, which eliminates the network eavesdropping threat.

**Message level authentication** using the WS-Security standard allows the passing of authentication tokens in a standard way by using SOAP headers. When two parties agree to use WS-Security, the precise format of the authentication token must be agreed upon. The following types of authentication token can be used and are supported:

- User name and password.
- Kerberos tickets.
- X.509 certificates.
- Custom tokens.

It is possible to send user names and password credentials in the SOAP header. However, because these are sent in plaintext, this approach should only be used in conjunction with SSL due to the network eavesdropping threat. Instead of sending a plaintext password, a password digest may be sent. However, unless this approach is used over a secure channel, the data can still be intercepted by attackers and reused to gain authenticated access to the web service. To help address this replay attack threat, a nonce and a creation timestamp can be combined with the digest. By sending the password digest with nonce and timestamp, the web service must maintain a table of nonce values and reject any message that contains a duplicate nonce value. While the approach helps protect the password and offers a basis for preventing replay attacks, it suffers from clock synchronization issues between the user and provider when calculating an expiration time, and it does not prevent an attacker from capturing a message, modifying the nonce value, and then replaying the message to the web service. To address this threat, the message must be digitally signed. The same way Kerberos tickets or X.509 certificates can be sent as security tokens.

In **application level authentication** a custom authentication mechanism can be designed by using custom SOAP headers for an application.

## **Authorization Strategies**

There exist two basic authorization strategies:

**Role based:** Users are partitioned into application-defined, logical roles. Members of a particular role share the same privileges within the application. Access to operations is authorized based on the role-membership of the caller. Resources are accessed using fixed identities, such as process identities. The resource managers trust the application to correctly authorize users and they authorize the trusted identity.

**Resource based:** Individual resources are secured using access control lists. The application impersonates the caller prior to accessing resources, which allows the operating system to perform standard access checks. All resource access is performed using the original caller's security context. This impersonation approach severely impacts application scalability, because it means that connection pooling cannot be used effectively within the application's middle tier.

In the majority of applications where scalability is essential, a role-based approach to authorization represents the best choice. For certain smaller scale intranet applications that serve per-user content from resources, such as files, that can be secured with access control lists against individual users, a resource-based approach may be appropriate.

### Resource Access Models

The two contrasting approaches to authorization can be seen within the two most commonly used resource-access security models used by web applications and distributed multi-tier applications:

- The trusted subsystem model and
- the impersonation/delegation model.

In the **trusted subsystem model**, the middle tier service uses a fixed identity to access downstream services and resources. The security context of the original caller does not flow through the service at the operating system level, although the application may choose to flow the original caller's identity at the application level. It may need to do so to support back-end auditing requirements, or to support per-user data access and authorization. The model name comes from the fact that the downstream service trusts the upstream service to authorize callers. The database trusts the middle tier to authorize callers and allow only authorized callers to access the database using the trusted identity. The pattern for resource access in the trusted subsystem model is the following:

1. Authenticate users.
2. Map users to roles.
3. Authorize based on role membership.
4. Access downstream resource manager using a fixed trusted identity.

Some resource managers may need to be able to perform slightly more fine-grained authorization, based on the role membership of the caller. It is for example possible to create two groups of users, one who should be authorized to perform read/write operations and the other to perform read-only operations.

With the **impersonation/delegation model** a service or component impersonates the client's identity before it accesses the next downstream service. If the next service in line is on the same computer, impersonation is sufficient. Delegation is required, if the downstream service is located on a remote computer. As a result of the delegation the security context used for the downstream resource access is that of the client. This model is typically used for the following reasons:

- It allows the downstream service to perform per-caller authorization using the original caller's identity.
- It allows the downstream service to use operating system-level auditing features.

As an example of this technique, a middle-tier enterprise services component might impersonate the caller prior to accessing a database. The database is accessed using a database connection tied to the security context of the original caller. With this model, the database authenticates each and every caller and makes authorization decisions based on permissions assigned to the individual caller's identity.

The trusted subsystem model is used in the majority of internet applications and large-scale intranet applications, primarily for scalability reasons. The impersonation model tends to be used in smaller-scale applications, where scalability is not the primary concern and those applications where auditing is a critical concern for reasons of nonrepudiation.

The primary advantage of the impersonation/delegation model is auditing close to the data. Auditing allows administrators to track which users have attempted to access specific resources. Generally auditing is considered most authoritative if the audits are generated at the precise time of resource access and by the same routines that access the resource. The impersonation/delegation model supports this by maintaining the user's security context for downstream resource access. This allows the back-end system to authoritatively log the user and the requested access. The disadvantages associated with the impersonation/delegation model include:

- *Technology challenges.* Most security service providers don't support delegation, Kerberos is the only exception. Processes that perform impersonation require higher privileges.
- *Scalability.* The impersonation/delegation model means that database connection pooling cannot be effectively used, because database access is performed by using connections that are tied to the individual security contexts of the original callers. This significantly limits the application's ability to scale to large numbers of users.
- *Increased administration effort.* Access control lists on back-end resources need to be maintained in such a way that each user is granted the appropriate level of

access. When the number of back-end resources and the number of users increases, a significant administration effort is required to manage those lists.

The trusted subsystem model offers the following advantages:

- *Scalability.* The trusted subsystem model supports connection pooling, an essential requirement for application scalability. Connection pooling allows multiple clients to reuse available, pooled connections. It works with this model because all back-end resource access uses the security context of the service account, regardless of the caller's identity.
- *Minimizes back-end ACL management.* Only the service account accesses back-end resources. Access control lists are configured against this single identity.
- *Users can't access data directly.* In the trusted-subsystem model, only the middle-tier service account is granted access to the back-end resources. As a result, users cannot directly access back-end data without going through the application and being subjected to application authorization.

The trusted-subsystem model suffers from a couple of drawbacks:

- *Auditing.* To perform auditing at the back end, it is possible to explicitly pass the identity of the original caller to the back end at the application level, and have the auditing performed there. The middle-tier needs to be trusted and there is a potential repudiation risk. Alternatively, an audit trail can be generated in the middle tier and then be correlated with back-end audit trails.
- *Increased risk from server compromise.* In the trusted-subsystem model, the middle-tier service is granted broad access to back-end resources. As a result, a compromised middle-tier service potentially makes it easier for an attacker to gain broad access to back-end resources.

### 7.3.3 Secure Communication

Many applications pass security sensitive data across networks to and from end users and between intermediate application nodes. Sensitive data might include credentials used for authentication, or confidential data. To guard against unwanted information disclosure and to protect the data from unauthorized modification while in transit, the channel between communication end points must be secured. Secure communication provides the following two features:

- *Privacy.* Privacy is concerned with ensuring that data remains private and confidential, and cannot be viewed by eavesdroppers. Privacy is usually provided by means of encryption.

- *Integrity.* Secure communication channels must also ensure that data is protected from accidental or deliberate modification while in transit. Integrity is usually provided by using MACs.

The most commonly used technology to secure the channel between clients and servers is Secure Sockets Layer/Transport Layer Security (SSL/TLS). Internet Protocol Security (IPSec) provides a transport level secure communication solution and can be used to secure the data sent between two computers, for example an application server and a database server.

When a web request flows across the physical deployment tiers of an application, it crosses a number of communication channels. The client-to-web server link may be over the internet or corporate intranet and typically uses HTTP. The remaining two links are between internal servers within the corporate domain. Nonetheless, all three links represent potential security concerns. The choice of technology depends on a number of factors including the transport protocol, end point technologies, and environmental considerations such as hardware, operating system versions and firewalls.

## SSL

SSL/TLS is used to establish an encrypted communication channel between client and server. When SSL is used the following facts must be taken into account:

- When SSL is applied, the client uses the HTTPS protocol and the server listens on TCP port 443.
- The application's performance should be monitored when SSL is used. SSL uses complex cryptographic functions to encrypt and decrypt data and as a result impacts the performance of applications. The largest performance hit occurs during the initial handshake, where asymmetric public/private-key encryption is used. After a secure session key is generated and exchanged, faster, symmetric encryption is used to encrypt application data.
- SSL requires a server authentication certificate to be installed on the servers.

## IPSec

IPSec can also be used to secure the data sent between two computers. IPSec is completely transparent to applications as encryption, integrity, and authentication services are implemented at the transport level. Applications continue to communicate with one another in the normal manner using TCP and UDP ports. Using IPSec it is possible to:

- Provide message confidentiality by encrypting all of the data sent between two computers.
- Provide message integrity between two computers without encrypting data.

- Provide mutual authentication between two computers.
- Restrict which computers can communicate with each other. It is also possible to restrict communication to specific IP protocols and TCP/UDP ports.

However IPsec is not compatible with any NAT based devices, particularly NAT based firewalls.

### Choosing Between SSL and IPsec

When choosing between IPsec and SSL the following points have to be considered:

- IPsec can be used to secure all IP traffic between computers; SSL is specific to an individual application.
- IPsec is a computer-wide setting and does not support the encryption of specific network connections. However, sites can be partitioned to use or not use SSL.
- IPsec is transparent to applications, so it can be used with secure protocols that run on top of IP such as HTTP, FTP and SMTP. However, SSL/TLS is closely tied to the application.
- IPsec can be used for computer authentication in addition to encryption. This is particularly significant for trusted subsystem scenarios, where the database authorizes a fixed identity from a specific application running on a specific computer. IPsec can be used to ensure that only specific application servers can connect to another server, in order to prevent attacks from other computers.
- SSL can work through NAT-based firewalls, while IPsec cannot.

### Message Level Encryption

In a closed environment where both endpoints are under control, SSL or IPsec may be used to provide transport layer encryption. In other environments and where messages are routed through intermediate application modes, a message level solution is required. The WS-Security standard defines a confidentiality service based on the World Wide Web Consortium (W3C) XML Encryption standard that allows the encryption of some or all of a SOAP message before it is transmitted.

## 7.4 Summary

RFID systems have to deal with huge amount of data and are therefore required to implement an operational data management architecture. Data is ideally preprocessed at the edge by concentrators. With pipelining a flexible and scalable system can be created.



An RFID system consists of hardware devices like readers, tags and printers on one hand and of the software components such as the middleware and the enterprise applications on the other hand. The middleware should ideally be capable of:

- Reader and device management,
- data management,
- application integration and
- scale well and be easy to administrate.

A typical middleware would thus consist of a central point of data access and a central point of configuration access. Part of the data and configuration control is moved forward to the tube managers. An architecture of this kind is particularly vulnerable at the central points of access because power is bundled there. The system is susceptible to:

- Parameter manipulation,
- eavesdropping and
- message replay.

The security mechanisms to choose depend largely on the infrastructure where the RFID system is deployed (see figure 7.4).

Security can be implemented point-to-point at the transport level, which is the mechanism of choice when transport mechanisms and both endpoints are under control. Alternatively in heterogenous webservice environments where endpoints and intermediate application nodes are not under direct control, message level, which is also called end to end security, can be implemented with Web Services Security (WSS).

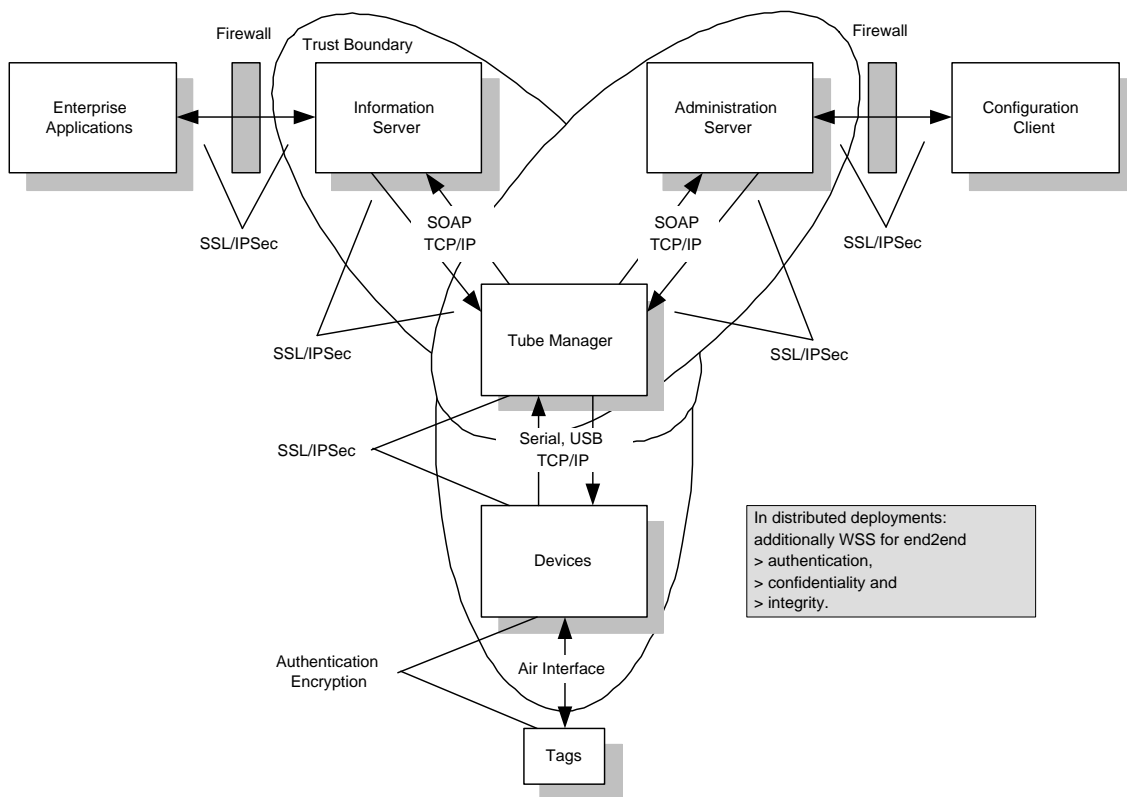


Figure 7.4: Secure Architecture for a Distributed RFID Infrastructure

# Chapter 8

## Conclusion

### 8.1 Summary

Radio Frequency Identification (RFID) uses radio transmission to recognize, categorize, locate and track objects. RFID systems consist of readers, tags and a back-end application or database for storage and management of the collected data. The tags are attached to items and can be read when they enter a reader's antenna field.

Depending on their requirements RFID systems need to satisfy the security services or a subset thereof. Integrity of RFID systems depends on the relation between tag and data, tag and object and tag and reader. This allows the several fraud scenarios to be identified. Security risks to RFID systems in the medium and long term, depend on the costs an attacker has to spent as well as the costs and efficiency of countermeasures. Current security mechanisms can provide reasonable protection at reasonable costs.

Privacy is a threat that primarily concerns clients of RFID systems, who use tags or objects that are identified by tags but have no control over the data stored on the tags. Privacy can be violated by the operator or by a third party. If an RFID system stores personal data the privacy of the passive party is threatened by eavesdropping or unauthorized access. This is called a threat to data privacy. Location privacy on the other hand is the protection from tracking of tags and reader interactions. There exist several suggestions to guarantee privacy in RFID systems. Data privacy is closely coupled to security issues and can be achieved with security mechanisms, whereas the suggested methods for achieving location privacy are often impractical. As of now the threat to location privacy is small but it will rise as we come closer to ubiquitous computing.

RFID systems handle huge amounts of data and are therefore based on an operational data management architecture. Data is preprocessed at the edge by concentrators. With pipelining a flexible and scalable system can be achieved. An RFID system consists of hardware devices and of software components. A typical middleware offers a central point of data access and a central point of configuration access. Part of the data and configuration control is moved forward to device managers. An architecture of this kind is particularly vulnerable at the central points of access because power is bundled there.

Security can be implemented point-to-point at the transport level or end-to-end at the message level. The first alternative is an option, when transport mechanisms and both endpoints are under control. In heterogenous webservice environments, where endpoints and intermediate application nodes are not under direct control, the second is the option of choice.

## 8.2 Outlook

The future of RFID is not solely determined by its technical possibilities but also by standardization, development of market and prices, information security, privacy and social acceptance. It is expected that by 2010 technical difficulties that are inhibitory for the large scale deployment of RFID, incomplete read rates, short ranges and difficulties with metal and liquids will be overcome. Due to the ongoing standardization process in RFID and the great expectations put in the EPC Gen2 UHF standard, prices are expected to fall in the following years. The development of prices over the next year will play a key role in the breakthrough of RFID since the costs are a significant decision factor in investment calculations. Costs are split into fixed costs which includes the RFID infrastructure, and variable costs such as tag costs. Combined with the expected cash flow an amortization time or a net present value, which is then can be calculated and compared to alternative investments.

Modern IT applications reveal a general trend towards increasing distribution, JIT networking and an associated rise in complexity. This phenomenon is also known as ubiquitous or pervasive computing. Due to distributed processing of actions of individual users, transparency is lost, making frauds and abuse harder to detect. However just transparency will play a decisive role in the social acceptance of RFID. It will therefore be a prime objective of companies to apply RFID in a transparent and secure way. This development will further necessitate adapted security solutions depending on the individual system requirements.

# Bibliography

- [Avo05] Gildas Avoine. Adversarial Model for Radio Frequency Identification. Cryptology ePrint Archive, Report 2005/049, 2005.
- [BAN98] M. Burrows, M. Abadi, and R. M. Needham. A Logic of Authentication. *Proceedings of the Royal Society of London*, 426:233–271, 1998.
- [BL04] Bharat K. Bhargava and Leszek Lilien. Vulnerabilities and Threats in Distributed Systems. In R. K. Ghosh and Hrushikesh Mohanty, editors, *ICDCIT*, volume 3347 of *Lecture Notes in Computer Science*, pages 146–157. Springer, 2004.
- [BM90] S. M. Bellovin and M. Merrit. Limitations of the Kerberos Authentication System. *Computer Comm. Review*, 20(5):119–132, 1990.
- [BP98] Giampaolo Bella and Lawrence C. Paulson. Mechanising BAN Kerberos by the Inductive Method, March 23 1998.
- [BP01] Steve Burnett and Stephen Paine. *RSA Security's Official Guide to Cryptography*. McGraw-Hill, New York, NY, USA, 2001. Includes CD-ROM.
- [Cen03] Auto-ID Center. *900 MHz Class 0 Radio Frequency (RF) Identification Tag Specification*. Auto-ID Center/EPCglobal, Cambridge, MA, USA, 2003.
- [DA99] T. Dierks and C. Allen. RFC 2246: The TLS Protocol Version 1, January 1999.
- [DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES the Advanced Encryption Standard*. Springer-Verlag, Berlin, Germany / Heidelberg, Germany / London, UK / etc., 2002.
- [ECM04] ECMA. *ECMA-340: Near Field Communication — Interface and Protocol (NFCIP-1)*. ECMA (European Association for Standardizing Information and Communication Systems), Geneva, Switzerland, December 2004.
- [EPC05] EPCglobal, 2005. <http://www.epcglobalinc.org> [July 2005].
- [FDW04] Martin Feldhofer, Sandra Dominikus, and Johannes Wolkerstorfer. Strong Authentication for RFID Systems Using the AES Algorithm. In Marc Joye and

Jean-Jacques Quisquater, editors, *Workshop on Cryptographic Hardware and Embedded Systems – CHES 2004*, pages 357–370, Boston, Massachusetts, USA, August 2004. IACR, Springer-Verlag.

- [Fel03] Martin Feldhofer. A Proposal for Authentication Protocol in a Security Layer for RFID Smart Tags, 2003.
- [Fin03] Klaus Finkenzeller. *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*. John Wiley and Sons, New York, NY, USA; London, UK; Sydney, Australia, second edition, 2003.
- [FK04] Thomas Finke and Harald Kelter. Abhörmöglichkeiten der Kommunikation zwischen Lesegerät und Transponder am Beispiel eines ISO14443-Systems. Technical report, Bundesministerium für Sicherheit in der Informationstechnik, 2004.
- [FKK96] Alan O. Freier, Philip Kariton, and Paul C. Kocher. The SSL Protocol: Version 3.0. Internet draft, Netscape Communications, 1996.
- [FRJ04] Kenneth P. Fishkin, Sumit Roy, and Bing Jiang. Some Methods for Privacy in RFID Communication. Technical report, 1st European Workshop on Security in Ad-Hoc and Sensor Networks (ESAS), 2004.
- [fSidI04] Bundesamt für Sicherheit in der Informationstechnik. *Risiken und Chancen des Einsatzes von RFID-Systemen*. SecuMedia, D-55205 Ingelheim, 2004.
- [Gar02] Simson Garfinkel. An RFID Bill of Rights. *Technology Review*, 2002.
- [Gra04] Jack Grasso. The EPCglobal Network: Overview of Design, Benefits and Security. Technical report, EPCglobal Inc., September 2004.
- [Gro03] Mifare Standardization Group. Mifare Application Directory, October 2003.
- [HH04] Steve Hodges and Mark Harrison. Demystifying RFID: Principles & Practicalities. Technical report, Auto-ID Center, January 2004.
- [HM04] Dirk Henrici and Paul Müller. Tackling Security and Privacy Issues in Radio Frequency Identification Devices. In Alois Ferscha and Friedemann Mattern, editors, *Pervasive Computing*, volume 3001 of *Lecture Notes in Computer Science*, pages 219–224, Vienna, Austria, April 2004. Springer-Verlag.
- [JRS03] Ari Juels, Ronald L. Rivest, and Michael Szydlo. The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. In Vijay Atluri and Peng Liu, editors, *Proceedings of the 10th ACM Conference on Computer and Communication Security (CCS-03)*, pages 103–111, Washington, DC, USA, October 2003. ACM, ACM Press.

- [Jue04] Ari Juels. Minimalist Cryptography for Low-Cost RFID Tags. In *The Fourth International Conference on Security in Communication Networks – SCN 2004*, Amalfi, Italia, September 2004. Springer-Verlag.
- [Kea03] A. T. Kearney. Meeting the Retail RFID Mandate. Technical report, A. T. Kearney Inc., November 2003.
- [KP04] Heiko Knospe and Hartmut Pohl. RFID Security. *Information Security Technical Report*, 9(4):39–50, November–December 2004.
- [Lea04] Sharyn Leaver. Evaluating RFID Middleware. Technical report, Forrester Research Inc., August 2004.
- [Lia04] Michael J. Liard. Radio Frequency Identification (RFID) Middleware Solutions: A Global Market Opportunity. Technical report, Venture Development Corporation, August 2004.
- [LSF04] Matthias Lampe, Martin Strassner, and Elgar Fleisch. A Ubiquitous Computing Environment for Aircraft Maintenance. In *ACM Symposium on Applied Computing 2004*, Nicosia, Cyprus, March 2004.
- [Mao04] Wenbo Mao. *Modern Cryptography: Theory and Practice*. Prentice-Hall PTR, Upper Saddle River, NJ 07458, USA, 2004.
- [Mur05] Jean V. Murphy. On the Edge: Understanding the RFID Framework. February 2005.
- [MW04] David Molnar and David Wagner. Privacy and Security in Library RFID: Issues, Practices, and Architectures. In Birgit Pfitzmann and Peng Liu, editors, *Conference on Computer and Communications Security – ACM CCS*, pages 210–219, Washington, DC, USA, October 2004. ACM, ACM Press.
- [NS78] Roger M. Needham and Michael D. Schroeder. Using Encryption for Authentication in Large Networks of Computers. *Communications of the ACM*, 21(12):993–999, December 1978.
- [NTR02] NTRU. GenuID, 2002. <http://www.ntru.com> [Feb 2005].
- [OAS04] OASIS. Web Services Security: SOAP Message Security 1.0, 2004.
- [Org96] International Standardization Organization, 1996. <http://www.iso.org> [July 2005].
- [Org04a] International Civil Aviation Organization. Biometrics Deployment of Machine Readable Travel Documents. Technical report, International Civil Aviation Organization, May 2004.

- [Org04b] International Civil Aviation Organization. Development of a Logical Data Structure - LDS for Optional Capacity Expansion Technologies. Technical report, International Civil Aviation Organization, May 2004.
- [Org04c] International Civil Aviation Organization. PKI for Machine Readable Travel Documents offering ICC Read-Only Access. Technical report, International Civil Aviation Organization, October 2004.
- [OSK03] Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita. Cryptographic Approach to "Privacy-Friendly" Tags. In *RFID Privacy Workshop*, MIT, MA, USA, November 2003.
- [Öst04] Bundeskanzleramt Österreich. Die österreichische Bürgerkarte. 2004. <http://www.buergerkarte.at> [July 2005].
- [Osw04] Elisabeth Oswald. Introduction to Information Security, 2004.
- [Pal04] Mark Palmer. Build an Effective RFID Architecture. Technical report, VeriSign, February 2004.
- [PD03] Larry Peterson and Bruce S. Davie. *Computer Networks: A Systems Approach*. Morgan Kaufmann Publishers, Los Altos, CA 94022, USA, third edition, 2003.
- [Sem03] Philips Semiconductors. Mifare DESFire. Technical report, September 2003.
- [She01] Tom Sheldon. *Encyclopedia of Networking and Telecommunications*. McGraw Hill, 2001.
- [Skl88] Bernard Sklar. *Digital Communications: Fundamentals and Applications*. Prentice Hall, 1988. **Review:** *IEEE Communications*, Vol. 27, No. 8, August 1989.
- [Sta04] Stefan Stadlober. Specification of a Runtime Environment for Multiple Application Infrastructures with the my-d Chip Family and the SPF Software. Technical report, Infineon Technologies AG, November 2004.
- [Tec04] Infineon Technologies. My-d Vicinity, February 2004.
- [Til01] James S. Tiller. *A Technical Guide to IPsec Virtual Private Networks*. CRC Press, 2000 N.W. Corporate Blvd., Boca Raton, FL 33431-9868, USA, 2001.
- [VB03] István Vajda and Levente Buttyán. Lightweight Authentication Protocols for Low-Cost RFID Tags. In *Second Workshop on Security in Ubiquitous Computing - Ubicomp 2003*, Seattle, WA, USA, October 2003.
- [Ver05] VeriSign. Securing RFID Data for the Supply Chain. Technical report, VeriSign, February 2005.



- [Wei03] Stephen Weis. Security and Privacy in Radio-Frequency Identification Devices. Master thesis, Massachusetts Institute of Technology (MIT), Massachusetts, USA, May 2003.
- [WSRE03] Stephen Weis, Sanjay Sarma, Ronald Rivest, and Daniel Engels. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. In Dieter Hutter, Günter Müller, Werner Stephan, and Markus Ullmann, editors, *International Conference on Security in Pervasive Computing – SPC 2003*, volume 2802 of *Lecture Notes in Computer Science*, pages 454–469, Boppard, Germany, March 2003. Springer-Verlag.