# Authentication issues in low-cost RFID

Ethmane El Moustaine

**THESE DE DOCTORAT CONJOINT**
**TELECOM SUDPARIS & L'UNIVERSITE PIERRE ET MARIE CURIE**

**Spécialité :**
**Télécommunications**

**Ecole Doctorale** : Informatique, Télécommunications et Electronique de Paris

**Présentée par**
**Ethmane EL MOUSTAINE**

**Pour obtenir le grade de**
**DOCTEUR DE TELECOM SUDPARIS**

**Doctorat conjoint Télécom SudParis & Université Pierre et Marie Curie**

# Problèmes liés à l'authentification dans les RFID à bas coût

**Soutenue le 13 décembre 2013, devant le jury composé de :**

| | | |
|---|---|---|
| M. Abdelmadjid BOUABDALLAH | Rapporteur | Université Technologique de Compiègne, France |
| M. Mohamed HAMDI | Rapporteur | Ecole Supérieure des Communications de Tunis, Tunisie |
| M. Jean-Jacques QUISQUATER | Examinateur | Université Catholique de Louvain, Belgique |
| M. Pierre SENS | Examinateur | Université Pierre et Marie Curie, France |
| M. Refik MOLVA | Examinateur | EURECOM Sophia Antipolis, France |
| Mme. Maryline LAURENT | Directrice de thèse | Télécom SudParis, France |

**Thèse n°2013TELE0030**

**PhD CONJOINT**
**TELECOM SUDPARIS & L'UNIVERSITE PIERRE ET MARIE CURIE**

**Specialty:**
**Telecommunications**

**Ecole Doctorale** : Informatique, Télécommunications et Electronique de Paris

**by**
**Ethmane EL MOUSTAINE**

**Defense hold on 13/12/2013 to obtain the degree of**
**DOCTEUR DE TELECOM SUDPARIS**

**PhD conjoint from Télécom SudParis & Université Pierre et Marie Curie**

# Authentication issues in low-cost RFID

**Dissertation defense committee:**

| | | |
|---|---|---|
| M. Abdelmadjid BOUABDALLAH | Rapporteur | Université Technologique de Compiègne, France |
| M. Mohamed HAMDI | Rapporteur | Ecole Supérieure des Communications de Tunis, Tunisie |
| M. Jean-Jacques QUISQUATER | Examinateur | Université Catholique de Louvain, Belgique |
| M. Pierre SENS | Examinateur | Université Pierre et Marie Curie, France |
| M. Refik MOLVA | Examinateur | EURECOM Sophia Antipolis, France |
| Mme. Maryline LAURENT | Directrice de thèse | Télécom SudParis, France |

**Thèse n°2013TELE0030**

## Acknowledgements

I would like to express my deepest gratitude to my supervisor Prof. Maryline LAURENT.
I thank her for giving me the opportunity to work under his direction, for encouraging me and believing in my capacities. I doubt that I will ever be able to convey my appreciation fully, but I owe her my eternal gratitude.

I would like to express my gratitude to Mr. Abdelmajid BOUABDALLAH and Mr. Mohamed HAMDI for accepting to review my dissertation and to be member of my PhD defense jury.

I gratefully acknowledge Mr. Jean-Jacques QUISQUATER, Mr. Pierre SENS and Mr. Refik MOLVA for accepting to be member of my PhD defense jury.

Last but not least, I would like to thank my family for their encouragements and their support.

Finally, I thank every person who helped me during this thesis.

# Summary

This thesis focuses on issues related to authentication in low-cost radio frequency identification technology, more commonly referred to as RFID.
This technology is one of the most promising technologies in the field of ubiquitous computing, it is often referred to as the next technological revolution after the Internet. However, due to the very limited resources in terms of computation, memory and energy on RFID tags, conventional security algorithms cannot be implemented on low-cost RFID tags making security and privacy an important research subject today.

First of all, we investigate the scalability in low-cost RFID systems by developing a ns-3 module to simulate the universal low-cost RFID standard EPC Class-1 Generation-2 in order to establish a strict framework for secure identification in low-cost RFID systems. We show that, the symmetrical key cryptography is excluded from being used in any scalable low-cost RFID standard due to the very short identification time for very large tag populations. That does not give any other alternatives than designing a lightweight approaches based on public key cryptography.

Then, we propose a scalable authentication protocol based on our adaptation of the famous public key cryptosystem NTRU. This protocol is specially designed for low-cost RFID systems, it can be efficiently implemented into low-cost tags, as tags are only required to implement a lightweight hash function, simple addition and bit-wise operations (circular shifts and xor operations).

Finally, we consider the zero-knowledge identification i.e. when the no secret sharing between the tag and the reader is needed. Such identification approaches are very helpful in many RFID applications when the tag changes constantly the field of administration.
We propose two zero-knowledge identification approaches based on GPS and randomized GPS schemes. The proposed approaches consist in storing in the back-end precomputed values in the form of coupons. So, the GPS-based variant can be private and the number of coupons can

be much higher than in other approaches thus leading to higher resistance to denial of service attacks for cheaper tags.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

The Radio Frequency Identification technology (RFID) has been a growing interest in the recent few years thanks to its unique features that allows passive smart label (RFID tag) to be scanned and identified with no need for visual or physical contact.

Contrary to popular belief, RFID technology is not new or recent, the first known application of RFID was the "friend or foe" identification system used in fighter planes in World War II, with a lead of 20 years on its wired equivalent, the smart card. However, the pervasive usage has started recently in biometric passports and as a practical replacement for optical barcodes reducing dramatically the costs in some production processes thanks to electronic manufacturing progress which makes low-cost RFID systems an economical replacement for optical barcode. Today, RFID technology is a common and useful tool in manufacturing, supply chain management, public transportation, physical access control, embedded medical devices, animal identification.

RFID systems are made up of three main components: RFID tag, RFID reader and back-end database as demonstrated in Figure C.1.

RFID tags can be classified into three types: active, passive, or semi-active. Passive tags are

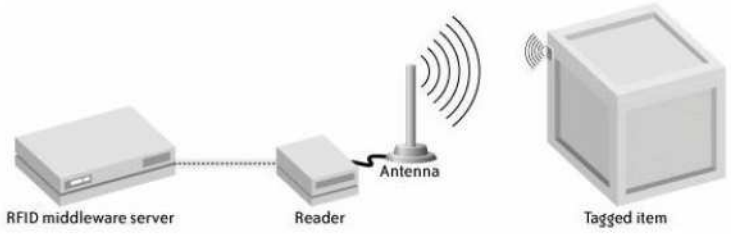


Figure 1.1: An RFID system [23]

those considered in this thesis, they are the most widely used and they do not contain any sort of on-board power; they are composed of a small microchip and an antenna. Tag's information can be read by an RFID reader, from some distance away and without requiring line-of-sight. The reader is used to provide energy to the tag by the magnetic resonance induction; therefore, as no energy is located on the tag, the tag is called a "passive" tag. The active tags are those which contain a battery from which they draw power for computation and for communication to readers (and even other tags). Semi-active tags (hybrid) contain an on-board battery that is exclusively utilized as a computation resource (and not for communication). This type of RFID is a combination of active and passive RFID technologies.
The reader powers the tag (passive and semi-active), retrieves and eventually transmits data related to tag to a back-end server for further processing.
The terminology "low-cost RFID tags" is widely used to design very low-cost passive RFID tags, and it is adopted throughout this thesis.

Today, the low cost of RFID, the huge economic stakes that present and ease of use, make this technology the next technological revolution after the Internet.

## 1.1 Authentication Challenges in Low-Cost RFID systems

The authentication is defined by the International Organisation for Standardisation [29] as a mean for confirming the identity of a claimed entity. In the information security, the authentication requires to use cryptographical algorithms which are known to require important resources in terms of computation, power and memory.
The identification is what happens when a prover claims to have a certain identity to the verifier. The difference with authentication is that authentication requires a proof that cannot be refuted of the claimed identity.
RFID, as its name suggests, the identification consists its very foundation. This identification must be secure to avoid the impersonation of tag and reader. However, to make RFID attractive for pervasive deployment (large scale deployment) two conditions need to be fulfilled. First, a very low-cost of tags should be guaranteed which leading to very constrained devices which cannot perform complex computations. Second, the scalability issues when tags massive scanning is required, need to be solved.
It is thus inevitable that, the disadvantage of RFID technology would be related to the security

and privacy risks thus making it more difficult to resist to standard attacks that are: replay attacks, man-in-the middle, denial of service, etc. With tags answering to any reader queries, privacy issues are made challenging and are related to clandestine tracking, location privacy, and forward secrecy. In a very simple way, an RFID identification protocol is vulnerable to clandestine tracking or location privacy when an adversary can recognize an RFID tag seen earlier. Forward secrecy is the property that a disclosed secret information about the tag does not reveal past secrets.

Today these issues are a major challenge for researchers, demanding to expand the boundaries of cryptography. These research works are encouraged by the European Commission. In 2008, a draft recommendation [15] was published on RFID security and privacy stating that RFID applications need to operate in a secure manner and that research needs to lead to high-performance and low-cost security solutions for RFID devices.

Another aspect related to RFID technology is the open system feature when the system interrogating the tag cannot be identified ahead of time due to the mobility aspect of RFID tags in some use cases i.e. there is no guarantee that the reader (verifier) is not malicious. In such context, solutions are zero-knowledge identification protocols as they do not require secrets sharing between the tag and the reader. However, these types of protocols as they are, require resources beyond capabilities of low-cost RFID tags.

The scheme commonly used to perform the authentication is called challenge/response: the reader sends a challenge $c$ to the tag that proves its identity by responding to this challenge. Obviously, an adversary should not be able to masquerade as the tag, even if he eavesdrops the previous answers of the tag. Respond to reader, consists in encrypting the received challenge $c$ using an encryption algorithm $E$, a random secret cryptographic key $k$ and eventually a fresh random value $r$ generated by the tag. This principle is illustrated in Figure 1.2.

| **Reader** | | **Tag** |
|---|---|---|
| choose $c \in_R \{0,\ 1\}^n$ | $\xrightarrow{c}$ | choose $r \in_R \{0,\ 1\}^n$, $E_k(c,\ r)$ |
| Retrieve $k$, check $E_k(c,\ r)$ | $\xleftarrow{r,\ E_k(c,\ r)}$ | |

Figure 1.2: Challenge/response identification scheme

However, if the encryption algorithm $E$ is symmetrical, this type of protocol has two shortcomings. One of them is its unscalability if the entire tag's response is different in each authentication attempt, in this case the system must do an exhaustive search to find the secret key $k$

to authenticate the tag.

The other one is the privacy, if an adversary can computationally distinguish a tag i.e make the link between a tag's response and a tag seen earlier, or if the tag's response is partially constant: for example, the tag responds with a supplementary $metaID(pseudo\ index)$ thus making easier for the system to search for the secret key, an attacker in this case can easily trace the tag between two $metaID$ updates.

The simultaneous provision of privacy and scalability in many low-cost RFID systems does not give other alternatives than designing lightweight approaches based on public key cryptography. Indeed, if the RFID system implements a public key cryptosystem, the reader has only one private key to decrypt any tag's response that can be different in each identification attempt. This solves the problems of scalability and privacy. Moreover, a high privacy level requires the use of some public key cryptographic techniques [52].

However, most of the researchers believe that the low-cost RFID cannot take advantage of public key cryptography and should therefore be based on symmetric cryptography. It is thus impossible under such assumption that an authentication protocol can support both scalability and privacy at low complexity cost.

## 1.2 Wireless attacks in RFID

The security and privacy in RFID systems mainly concern the wireless communications between tag and reader while the communications between the back-end and the reader are assumed as secure because the reader and back-end are devices with computing and battery resources, so there are no impediments to secure their in-between communications with strong symmetric or asymmetric key algorithms.

The attacker is assumed behaving according to the Dolev-Yao model, i.e. having full control over the wireless channel to replay, modify and store exchanged messages.

- Malicious Tracking: it is often referred to as the major privacy threat, the goal of the attacker is to trace a tag i.e. recognize a tag previously seen. In RFID systems, tag responds to any reader's query, so when tag answers to reader queries are unique, an attacker can track the movement of the tag owner. This is one of the main problems that ubiquitous computing has to solve as Mark Weiser already predicted in 1991 [54].

- Replay attacks: the attacker uses a simple method of exploiting a captured traffic ex-

changed between tag and reader and resends this traffic to be authenticated as a legitimate tag or reader.

- Man in the middle attacks: the attacker tries to obtain information from a legitimate tag or reader in order to impersonate it later.

- Denial of Service attacks: these attacks concern mainly the desynchronization between the tag and the reader/back-end. The attacker can make inconsistent a shared secret key between the tag and the reader in order that they cannot recognize each other in the future authentication sessions.

## 1.3 Contributions

In this thesis, we investigate the issues of scalability, privacy, and zero-knowledge in low-cost RFID systems using lightweight public key cryptography. Our contributions in this thesis are:

- In the first place, we investigate the scalability criterion by developing a simulator module for ns-3 that predicts the identification performances in many scenarios for the universal standard for low-cost RFID systems, the EPC Class-1 Generation-2.
  The specifications of this standard do not provide any security mechanism on tags which are just considered as radio-frequency barcodes. Our main objective is to obtain some key aspects such as the identification time for different tag populations. This enables getting for the first time realistic results and establishing a strict framework for secure identification in such low-cost RFID systems for different tag populations.

- In the second place, we propose a new scalable lightweight asymmetrical mutual authentication protocol especially designed for low-cost RFID systems. This approach is based on an adaptation that we have introduced on NTRU public key cryptosystem for the tag by bringing back the encryption process from a difficult mathematical operation, polynomials multiplication over ring, to only simple circular shifts and additions.

- In the third place, we investigate the zero-knowledge identification (no secret sharing between the tag and the reader) in low-cost RFID systems and we propose two private lightweight storage-security trade-offs based on GPS and randomized GPS public key schemes. Our approaches have two key benefits. First, the storage overhead is supported by the back-end server and not by the tag, so the GPS-based variant can also be private,

and the RFID system is less vulnerable to denial of service attacks than other approaches. Second, for authenticating to the reader, the tag only needs simple integer operations, so implementation can be done in less than 1000 gate equivalents (GEs).

In the course of the thesis, we have given substantial contributions to the French-German project RESCUE-IT. From a modelling perspective, RESCUE-IT represents a complete supply chain in the public security area and integrates security requirements. It also proposes a database about risks and mitigation procedures related to supply chains. From a requirements perspective, the project identifies the relevant security parameters, which need to be monitored and controlled. From an Internet of Things perspective, the project enhances the usage of wireless sensor networks and of RFID systems, tailored for the specific needs of a secured supply chain.

## 1.4 Thesis organization

The reminder of this thesis is organized in three chapters.

In the next chapter, we consider the simulation problem in RFID systems, we introduce works found in the literature related to simulation in RFID technology. We also introduce the fundamental concepts of EPC Class-1 Generation-2. Then, we present our module for ns-3 to simulate RFID systems and results from different scenarios.

In the third chapter, we consider the scalability in low-cost RFID systems and we show the limits of various scalable authentication protocols found in the literature, and we propose a new scalable lattice-based authentication protocol specially designed for low-cost RFID systems.

In the fourth chapter, we consider the importance of zero-knowledge identification in some RFID applications and we introduce a rapid state of the art pertaining to this concept. Then, we present our storage-security trade-off identification schemes for low-cost RFID systems.

In the last chapter, we conclude this thesis and we give the main future work directions.

# Chapter 2

# Massive reading simulation for low-cost RFID tags

EPCglobal Class-1 Generation-2 UHF Radio Frequency Identification standard, commonly known as the EPCGen2 for passive RFID technology is a new standard which was approved in 2004. The EPCGen 2 tag is considered as a successor to the barcode, it is a powerful mechanism for object identification. The EPCGen2 is widely accepted as the universal standard for low-cost RFID tags. It is designed to satisfy the supply chain requirements. The small size of EPCGen2 tags allows them to be implanted within objects and identification by frequency allows tags not only to be read in large numbers but also through visually or environmentally challenging conditions.

However, EPCGen2 doesn't define exactly how the reader (interrogator) should react in critical situations like collisions or how to manage different commands (Query, QueryRep and QueryAdjust) during the inventory round. This is why a simulator can be a very useful tool to discover the protocol behaviors in different scenarios.

In this chapter, we deal with this approach, we create a new module in Network Simulator ns-3 to predict the identification performances for EPCGen2 communication protocol in different configurations. This module allows to test the scalability of this standard in order to establish a strict framework for secure identification, for example, what kind of security approaches that could be envisaged. Moreover, RFID simulation is very useful for many research challenges such as security, privacy and throughput optimization over the tag-reader channel which are so much difficult to be tested in practice because of the integrated nature of RFID tags.

This module is able to simulate various realistic scenarios and configurations to predict the identification performance in many scenarios like check out control, reception control, expedition control, and inventory that serve to evaluate the performances of this standard in the supply chain for example.

We give some key aspects of different configurations like identification time for tag populations, statistics on collisions during the identification procedure to improve performances, etc.

The remainder of this chapter is organized as follows. We first introduce in Section 2.1 works related to RFID simulation and we demonstrate their limits. Section 2.2 then describes the EPCGen2 communication protocol. Sections 2.3 and 2.4 introduce our simulation model and ns-3; respectively. Section 2.5 describes our simulator. The simulated scenarios and their results are given in sections 2.6 and 2.7; respectively, before conclusions in section 2.8.

## 2.1 Related works

The simulation of RFID systems with conventional wireless network simulation tools (ns-2, ns-3, Omnet++, rtc.) involves the development of a new module that reflects the reality of RFID systems.

In [6], the authors provide a simulation of RFID under Network Simulator ns-2. The simulation is based on the available ns-2 802.11 already existing in ns-2, so this simulation does not reflect the reality of RFID systems, specially the adopted mac and physical layers are too much different than layers of RFID readers and tags.

In [12], the authors develop with Java an evaluation tool PETRA for ISO 18000-3 protocol. The developed software simulates the scenario of a group of tags that get into and get out the antenna reader's area at two different times. The tool gives the number of successfully identified tags and identification duration according to the number of tags and velocity. However [12] is inaccurate because it does not model the physical layer of RFID devices, so many fonctions are missing like signal propagation, capture, antenna directivity, backscatter, and tag mobility model.

In [18], authors propose a simulator RFIDSIM the implementation is done with Jist(Java in simulation time). RFIDSim implements the ISO 18000-6C communication protocol and supports pathloss, fading, backscatter, capture, and tag mobility models. RFIDSIM was essentially designed to facilitate the relative comparison of different medium access protocols and transmis-

sion control strategies of ISO 18000-6C. We identify several weaknesses in simulator [18]. It is unable to predict identification performances. Moreover it does not use a standard simulation platform like ns-2, ns-3 or Omnet++, is made of several software components, and as such is difficult to deploy. The architecture of the simulator is not clear enough for bringing modifications to it.

## 2.2 EPC Class 1 Generation 2 communication protocol

### 2.2.1 Generalities on EPC Gen2

The EPCGen2 standard [2] defines the physical and logical requirements for passive RFID tags in order to replace the traditional barcode. It was designed by EPCGlobal Inc. one of the leaders in the development of industry global wide standards for the EPC to support the use of RFID.

EPCGen2 reader talks first and tags can be read and write an infinite number of times contrary to previous EPC generations. The reader transmits information (commands) to tags by modulating a radio frequency signal.

Tags which are passive, receive both transmitted information and operating energy from the radio frequency signal: they respond to reader by modulating a continuous radio frequency wave signal transmitted by the reader itself.

Table 2.1 sums up the EPCGen2 characteristics of reader and tag.

| | |
|---|---|
| Operating Frequency | Ultra High Frequency range: 860 MHz - 960 MHz |
| Range | Approximatively 5 meters |
| Data rate | Forward link: 26.7 - 128 kbps |
| | Backscatter link: 5 - 640 kbps |
| Encoding | Forward link: pulse interval encoding (PIE) |
| | Backscatter link: FM0, Miller-modulated subcarrier |
| Modulation | Forward link:DSB-ASK, SSB-ASK or PR-ASK |
| | Backscatter link: ASK or PSK |
| Multiple access protocol | Variant of slotted Aloha (Q Protocol) |
| Tags | Passive, EPC, reserved and user memory |

Table 2.1: Characteristics of EPCGen2 protocol

### 2.2.2 EPCGen2 medium access protocol

In order to minimize collisions during Inventory round (identification process), the access of EPCGen2 tags to the shared wireless channel is based on a variant of slotted Aloha called Q Protocol.

In Aloha, the shared wireless channel is accessible to all stations. An active station, which has a message for another station broadcasts it directly. The main problem to solve is that of collisions. Indeed, in such environment there is no central control and transmission times of the messages are unpredictable. Slotted ALOHA is a refinement over the pure Aloha, time is divided into lots which has an impact on the performances. Any station can transmit whenever she wants, but it must start its transmission at the beginning of a slot.
In EPCGen2 medium access protocol, the Q parameter is specified in the Query command and ranges from 0 to 15 inclusive. To reply within the inventory round, each tag that receives the previous command from the reader, chooses a random number in the range 0 to $2^Q$-1 inclusive. Tags that get zero should reply immediately by a 16-bit pseudo random sequence (RN16) message. Otherwise, tag should wait for other commands.
There are three possibilities:

- No tags reply: the reader may send another Query, a QueryRep or a QueryAdjust command.
- One tag replies: the reader may identify the tag if every thing is going well.
- Multiple tags reply: the reader receives several RN16 messages up to the number of tags having zero value in their slot counter. The reader has to resolve the collision and sends an ACK. Otherwise, if the collision can not be solved, the reader issues a QueryRep or QueryAdjust command.

If no tag replies or many tags replay, the reader may send a QueryAdjust to solve the problem by modifying the initial Q value sent in the Query command by decreasing the Q value when the reader has no response to rise the probability to get a reply from a tag and it increases the Q value when there is more than one reply (to avoid the collisions problem). Figure 2.1 gives an algorithm for managing Q parameter according to the situation (C is a float ranging from 0.1 to 0.5). The reader manages tag populations using three basic operations which are: Select, Inventory and Access. We focus on the Select and the Inventory procedure because those ones

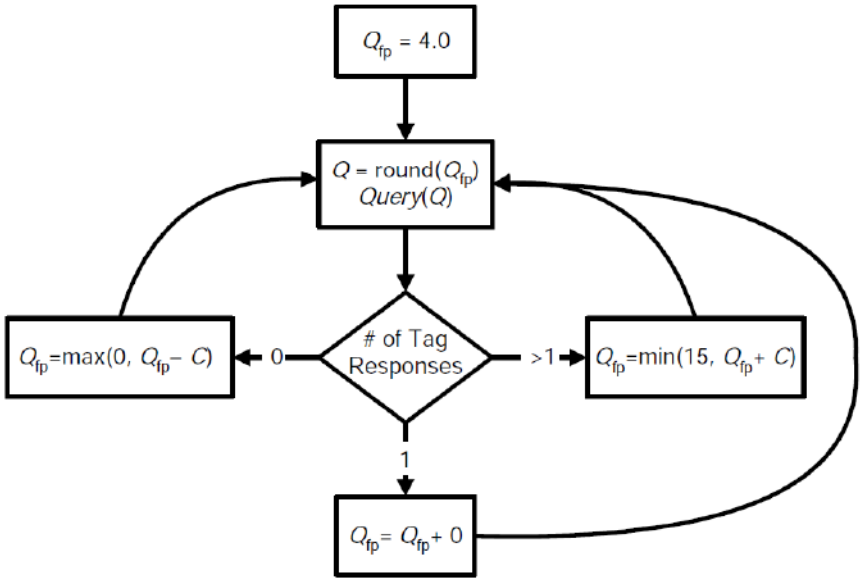


Figure 2.1: Example of algorithm for managing Q parameter [10]

are affecting the exchange duration and that's what we deal with in this chapter. For the access procedure, it is optional and it can be added in future works.

The reader supports four sessions $S0$, $S1$, $S2$, and $S3$. The tag has to participate in one and only one session during the inventory round. It has two states $A$ and $B$, and when the tag is powered up it inverts its inventoried flags ($A \rightarrow B$ or $B \rightarrow A$).

### 2.2.3 Select procedure

In the select procedure, the reader selects tags for Inventory or Access round. The reader applies Select command successively to select a particular tags population referring to a user-based criteria. Reader can send many Select commands at a time to perform set operations like unions, intersection and negation on the tag populations. Following this command, concerned tags will modify the selected flag or the inventoried flag for one of the four sessions. The criteria for determining whether the tag is concerned by the select command is when the mask sent in a command matches the one saved in the tag memory. For example as in the figure 2.2, we have one reader and a population of tags coming from two different constructors (X and Y) and those tags are all in the S1 session (one of the four sessions). In the inventory operation, we want only to communicate with tags from X constructor. So the reader will send at first a select command with S1 session and a mask with the serial of the X constructor to change the inventoried flag to "A" and another select command with S1 session and a mask with the serial of the Y constructor to change the inventoried flag to "B".

Therefore, in the inventory stage, reader will communicate only with tags in session S1 and with "A" inventoried flag.

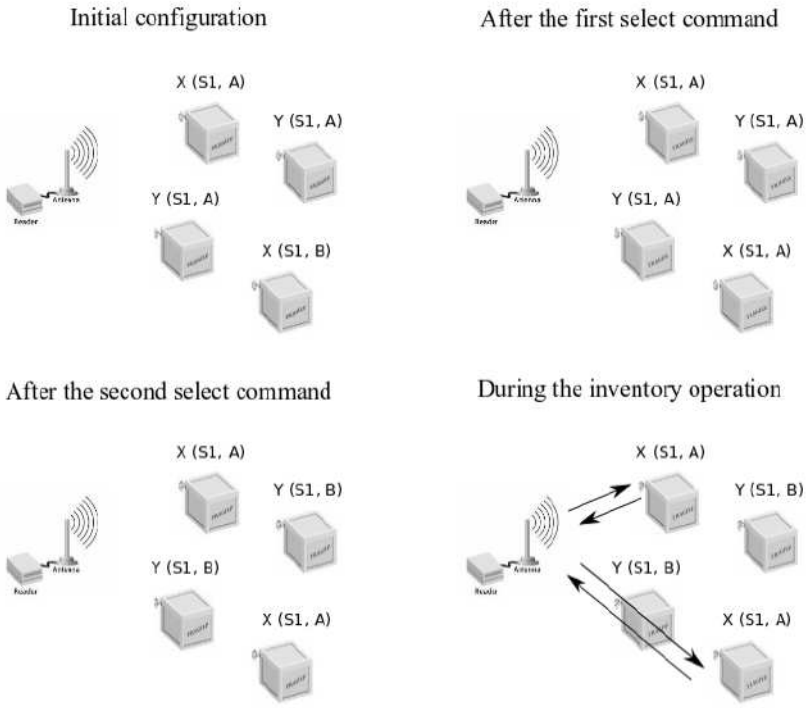


Figure 2.2: Select command's example

### 2.2.4 Inventory procedure

Inventory is the process by which a reader identifies tags i.e. reads the EPC of all tags. The inventory command set includes the following commands: Query, QueryRep, QueryAdjust, ACK and NAK:

- Query: the reader launches a new inventory round. We configure some parameters that define, for example, if the reader will communicate with tags according to their session and the inventoried flags or their selected flags. Another important parameter on this command is the Q parameter which is chosen randomly and sent to tags to avoid collisions i.e. differentiate response time of each one.

- QueryRep: this command is the most commonly used. Upon reception, the tag decreased the slot counter by one in order to get zero and sends RN16 message.

- QueryAdjust: the reader launches a new inventory round using the same parameters of the previous round except the Q parameter that is higher or smaller if there is a collision detection or if there is no tag response, respectively.
- ACK: the reader sends this response to acknowledge that it received an RN16 and echoes the RN16 value of the received message.
- NAK (Not-an-ACK): the reader sends a NAK message to inform a tag that its EPC was not successfully received.

The standard mentions that at least one select command should precede an inventory round. The tag passes through several states and transactions during the identification process as illustrated in Figure 2.3 The inventory round is started by a Query command in which the reader

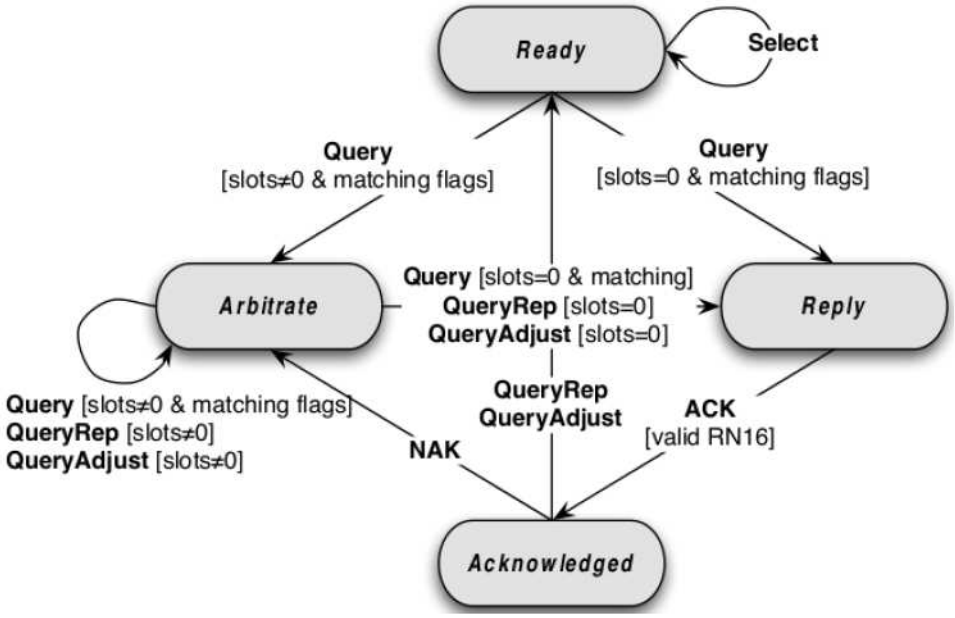


Figure 2.3: Tag state diagram of the EPCGen2 protocol

specifies the Q parameter and which tags participate in this round based on their inventoried flags and selected flags. When the tag receives this command, it chooses a random value strictly lower than $2^Q$ and transits to the reply state if the chosen random value is equal to zero. Tags that have generated a non-zero value must transit to arbitrate state and wait for Query, QueryRep or QueryAdjust command. As described in Figure 2.4 , in the case only one tag replies, the identification protocol works as follows:

- Tag generates and sends to reader a 16-bit random number RN16 (Random number range from 0 to FFFF inclusive) and transits to reply state.

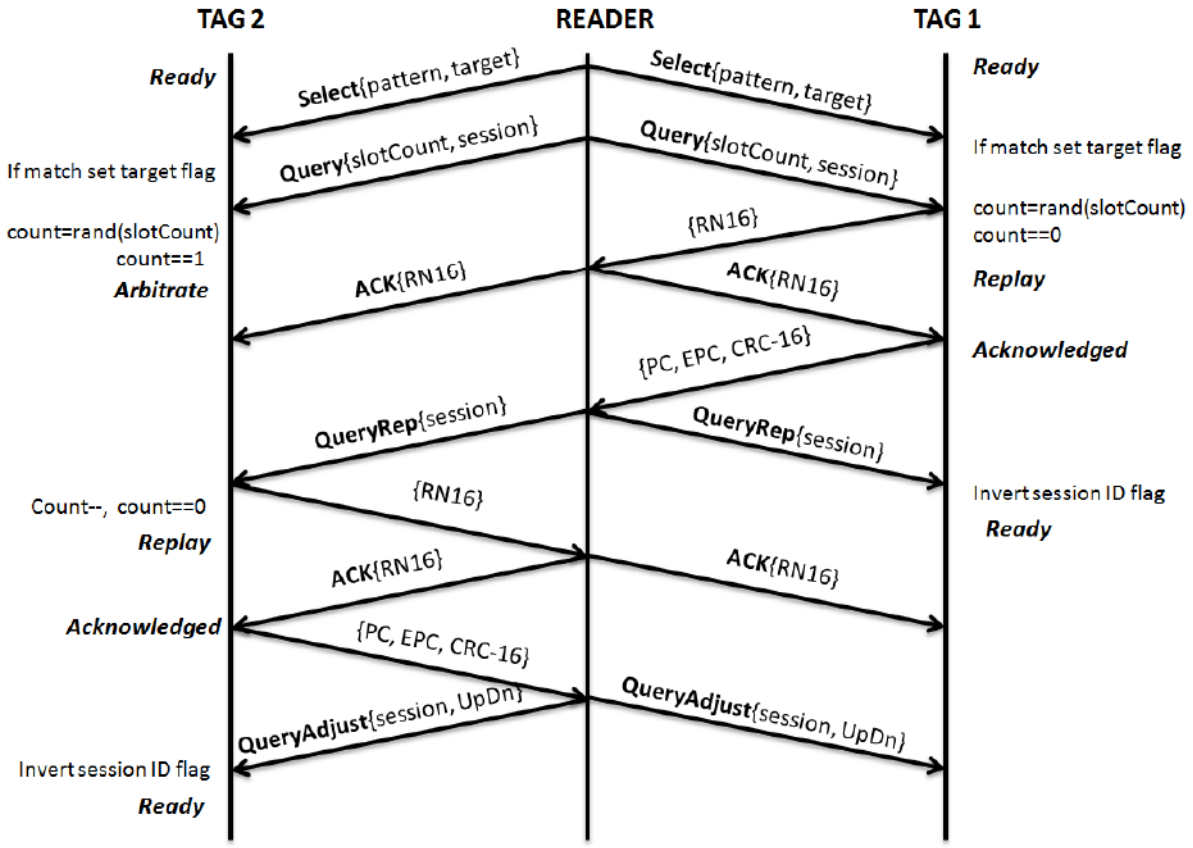


Figure 2.4: Tag and reader (interrogator) exchange steps supporting identification

- Reader receives the message RN16 and acknowledges the tag with an ACK containing the same RN16 value.
- The acknowledged tag changes its state to acknowledge, calculates and sends back to reader the EPC (tag's identifier), PC (Protocol Control) and CRC16 (16-bit Cyclic Redundancy Check) in one message.
- If the EPC is correctly received, the reader issues a QueryRep or QueryAdjust command. The identified tag changes its inventoried flag and returns to ready state, so it is no longer responding to any QueryRep or QueryAdjust commands. On the other hand, another tag will enter in reply state and response to reader.

Figure 2.4 presents only the main steps during the exchange operations between the reader and two tags in EPCGen2 protocol. The reader can interact more with the tag after the recovery of its identifier EPC for example with the aim of writing some information or killing the tag. Here, what we are concerned with is the most used functionalities of EPCGen2, i.e. the recovery of tag populations identifiers (EPCs).

### 2.2.5 Link timing performance

In EPCGen2, the exchange between the tag and the reader must follow very specific constraints on time as shown in Figure 2.5.

The parameter T4 is defined as the minimum time between reader commands, in Figure 2.5, T4 presents the waiting time between select and query commands. After Query or QueryRep or QueryAdjust, the reader should wait at least for T1 to receive a message from a tag, otherwise, it has to retransmit one of the previous commands. Even the tag has to wait T2 after sending message to receive response from the reader, otherwise, it returns to a previous state either ready or arbitrate in accordance with its actual state. The last timing parameter is T3 which is the time that a reader waits for after T1, before it issues another command; it's the tolerance margin added to T1 parameter to avoid the retransmission of another command.

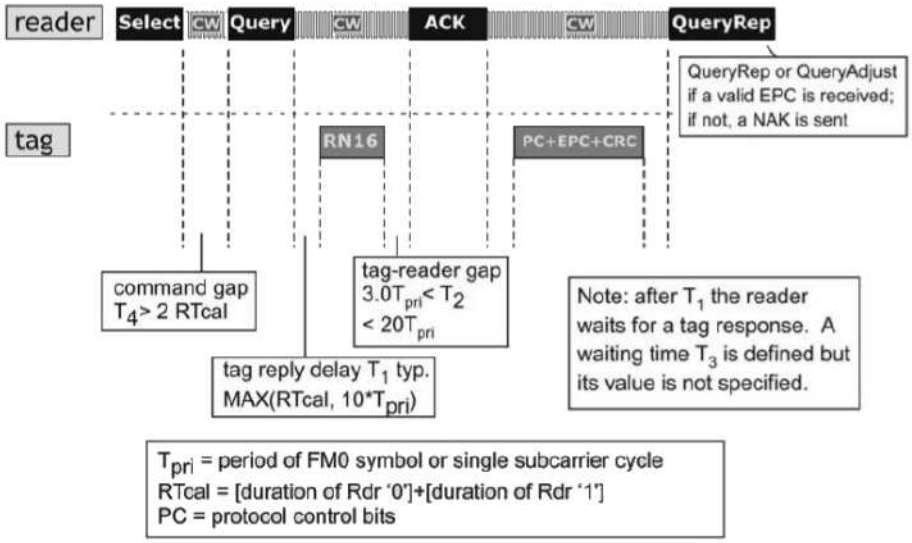


Figure 2.5: EPCGen2 timing constraint [11]

The time constants above (T1, T2, T3, and T4) are dependent on a parameter named Tari which is the reference time interval for reader-to-tag signaling, and it is the duration of a data-0. The configuration of this parameter determines both Data-0 and Data-1 length used in message's transmission, as shown in Figure 2.6, and, timeout parameters. Tari value as defined in the standard, should be between 6.25 $\mu$s and 25 $\mu$s [2].

## 2.3 Simulation model

To make the simulation quite realistic, we need to model the mobility and powering of tags, signal propagation between tags and reader, reception and transmission of the signals at the

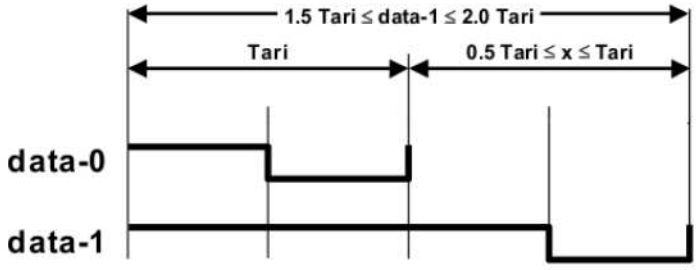


Figure 2.6: Data-0 and Data-1 parameters [2]

RFID readers and tags, commands specified in EPCGen2 communication protocol, etc. Regarding timing parameters and modulation, we adopt a model similar to that of [18] as described in Table 2.2.

| Parameter name | Value |
|---|---|
| Tari | 8.33 $\mu$s |
| Data-1 | 16.66 $\mu$s |
| TRext | 0 |
| Divide Ratio | 8 |
| Modulation | 2 |
| T1 | 70.7 $\mu$s |
| T2 | 18.7 $\mu$s |
| T3 | 62.5 $\mu$s |
| T4 | 50.0 $\mu$s |

Table 2.2: Timing parameters

In the most commonly used model in the free space [36], the received signal is attenuated by a D factor which presents the pathloss:

$$D = \left(\frac{4\pi r}{\lambda}\right)^2$$

Where r is the distance between receiver and transmitter and $\lambda$ is the wavelength. Many other configurations about power, sensitivity and other aspects are shown in table C.3.

We propose two algorithms for commands management and collisions that define the reader behavior in many situations. Indeed, the standard just defines general features and it leaves the freedom of choice for the implementation in order to get better performances according to each scenario, it doesn't point out explicitly how the reader addresses the problem of collisions and what command should be used in every situation.

| Parameter name | Value |
|---|---|
| **Reader** | |
| Frequency | 866 MHz |
| Power (EIRP) | 2000 mW |
| Antenna beamwidth | 60° |
| Sensitivity | -80 dBm |
| **Tag** | |
| Sensitivity | -14 dBm |
| BackScatter factor | 0.25 |
| Speed | 1 m/s |
| **Propagation channel** | |
| PathLoss exponent | 2.0 |

Table 2.3: Configuration parameters

### 2.3.1 Commands management algorithm

This algorithm is to speed up the identification during the inventory round. As demonstrated in Figure 2.7, it specifies how the reader should use the following inventory commands: Query, QueryRep and QueryAdjust. First, the reader starts the inventory round by sending a Query

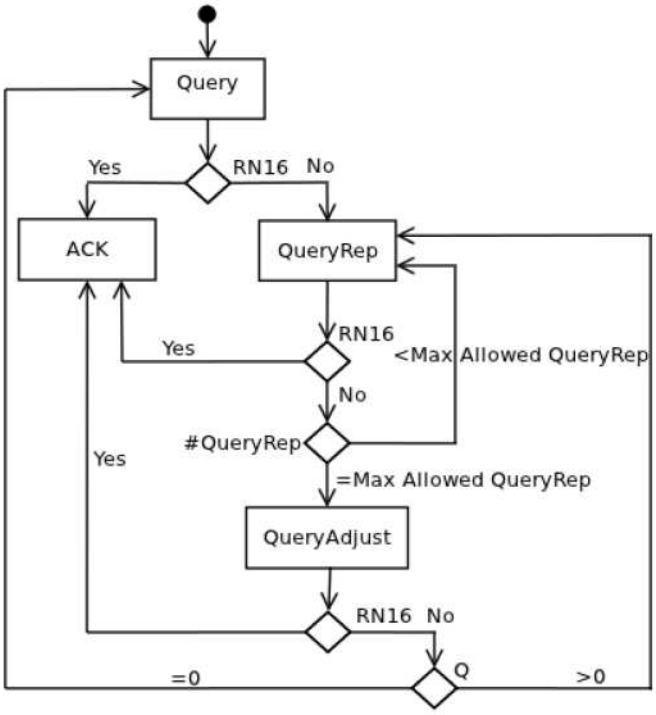


Figure 2.7: Activity diagram of Commands management algorithm

message. If a tag responds i.e. a RN16 message is get back, the exchange procedure continues with the tag, otherwise a number of QueryRep commands is sent in order to decrease the slot counters of tags. In case of no response, the reader sends a QueryAdjust with smaller value of Q that decreases more likely the new selected slot counters of tags in order to respond faster this

time. The sending of QueryRep and QueryAdjust continues until Q equals zero. If the reader gets no response back, either there is no tags in the antenna range or new tags get into the area after the last Query command and should receive a new one to be able to respond. Thus, the reader sends a Query command and a new inventory round can start again.
The reader stops the transmission after all tags are identified or several Query have been sent with no response back.

### 2.3.2 Anti-collision algorithm

During an inventory round, a collision can happen when more than one tag choose the same slot counter and reach zero value together so they respond with two RN16 messages at the same time.

As described in Figure 2.8, the command QueryAdjust increases the initial Q, so tags select other slot counters according to the new Q (slot counter $\in$ $[0,\ 2^Q - 1]$). Therefore, the probability that two tags choose again the same slot counter and respond at the same time is decreased. In this case, tags take more time to reach zero value, so the number of QueryRep increases to be sent after. The number of QueryRep is adaptive in order to avoid collisions and reduce the identification time as demonstrated in Figure 2.8.

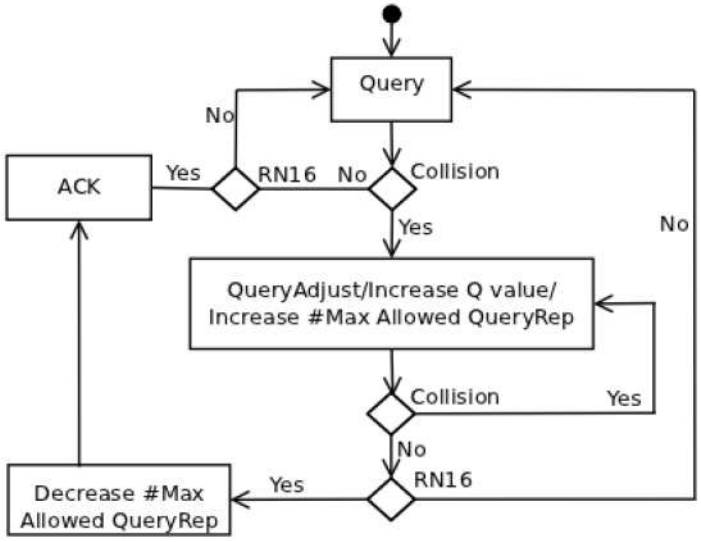


Figure 2.8: Activity diagram of anti-collision algorithm

## 2.4 Network Simulator 3

The Network simulator 3 (ns-3) is a next generation simulator designed to replace ns-3. It is a free, open source licensing (GNU GPLv2) and a discrete event network simulator. The simulation core and models are written in C++. ns-3 uses the waf build system which is a Python-based framework for configuring, compiling and installing applications.

As demonstrated in [53], ns-3 has better performance in terms of run time and memory usage over other network simulators (OMNet++, Jist and SimPy) and even ns-2.
ns-3 seems to be a good choice to implement EPCGen2 module for both performances and scalability.

## 2.5 The RFID module for ns-3

There is a huge difference between the existing modules in ns-3 and RFID, essentially in the physical layer and propagation channel, but many functionalities such as mobility, packet class, buffer, simulator, etc. are native to the ns-3 core.

### 2.5.1 Software Design

We model the physical and the logical layer of EPCGen2 communication protocol. Figure 2.9 shows a simplified version of the class diagram because every class has a lot of attributes and methods and could not be properly presented. Most used methods are described in Annex A.

#### 2.5.1.1 RFID Channel

The RFIDChannel extends the Channel class provided by ns-3, presents the radio propagation field. It is responsible for modelling signal propagation and broadcasting information to all the devices existing in the equipment's antenna range (tags and/or reader). It includes also propagation delay due to the distance between devices and power loss as a result of pathloss phenomena. It knows the references of all tags and readers. So it decides, referring to the device position and signal strength, which radio node will receive the message and which not.

One of the most important method in RFID Channel is send() which verifies whether the receiver is in the antenna's range or not, it calculates the propagation delay according to sender mobility and receiver mobility, it calculates the power loss due to the pathloss and it broadcasts the message to be received by all the equipments in the antenna's range.

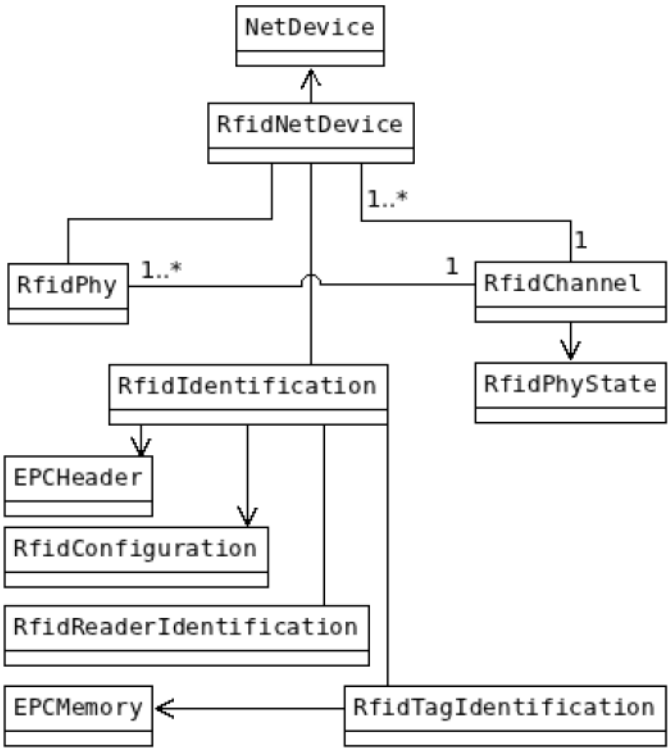


Figure 2.9: Class diagram of the proposed RFID module

#### 2.5.1.2 Physical layer

RFIDPhy extends the Object class provided by ns-3. It implements the physical layer of the RFID tags and readers. This class is responsible for transmitting and receiving messages from the field entity (RFID Channel), detecting collisions and delivering successfully received messages to the upper layer which is Identification entity.

When the Physical layer receives a message from the Channel by StartRecv(), it calls RfidPhyState class which saves the actual state of physical layer. His state could be idle, sending or receiving. So, the message is accepted only in idle state and it is rejected in sending or receiving state. A probable collision is declared and the higher layer is notified of the situation.

#### 2.5.1.3 Identification layer

RfidIdentification class is the logical entity that contains the logic of simulated devices. Its main purpose is to provide the interface to the device functionalities. This class is created to be the superclass for the Tag Identification layer and the Reader Identification Layer as there are common functionalities. All identification layers are attached to RfidConfiguration class and EpcHeader class. The first one is used to save and load device configurations like modulation

type, bit length, timeout before retransmission or changing device state, etc. The second one serves for creating packet and reading received packets.

RFIDTagIdentification contains the logic of simulated tags by providing all functionalities that simulate the tag working such as save and load reader configurations like modulation type, bit length, timeout before retransmission or changing reader state, etc. and also creating packet and reading received packets. For example, *Receive ()* method receives messages from the physical layer, transfers them to the *SetEquipementState ()* method which decides in the first step according to its state, if it is able to respond or not.
Then, if it does, it reads the message contents, switch state and generates a message as a response to the reader request.
When the message does not match what the tag is waiting for, it will return silent waiting for another command.

This class is connected to EpcMemory Class which presents the different memory banks of the tag. It is called in some cases, for example when the tag receives a Select command to verify if it matches command parameters and also when the tag sends the EPC identifier, it creates the response from Protocol Control, EPC identifier and CRC16 and all of these parameters are saved into the EPC bank.

RFIDReaderIdentification contains the logic of simulated reader. Its main purpose is to provide the interface to the reader functionalities such as save and load reader configurations like modulation type, bit length, timeout before retransmission or changing reader state, etc. and also creating packet and reading received packets. It provides all the functionalities that simulate the reader operations. Conforming to the standard, the reader starts talking first by sending the first message which can be Select or Query message. Also, the decision to respond to received messages is made by SetEquipmentState() method. After every sent message, a countdown is launched. If the reader doesn't receive a message before timing out, SendQueryRepOrAdjust() method is automatically called to send a Query, a QueryRep or QueryAdjust message in accordance with reader state.

#### 2.5.1.4 RFID network device

RFIDNetDevice class extends the NetDevice class provided by ns-3. NetDevice class is used in other modules like wifi and wimax as a network card which can be plugged in an IO interface of a device. For the RFID module, it's similar to the network device, so it contains, for every

device, its whole configuration for different layers in order to facilitate the access to different public methods from one layer to another for the same device.

## 2.6 Simulated scenarios

We simulate and study the performances of EPCGen2 communication protocol in three different scenarios in order to get some statistics on identification duration for variant tag populations, number of collisions detected over the inventory round, and power attenuation during exchange with moving tags. In practice, these scenarios correspond to the main steps in the supply chain (Inventory, reception/expedition control) for which the EPCGen2 is primarily designed.
Each of the azimuth beamwidth and the elevation beamwidth of the reader's antenna is 60°.

### 2.6.1 RFID fixed tags scenario

This scenario corresponds to the supply chain inventory and it is characterized by:

- An RFID reader that is permanently sending Query, QueryRep and QueryAdjust in order to identify all tags in the reader antenna's range.
- A tag population randomly distributed as in a pallet with a size of 1m x 1m x 1m, 2 meter far from reader's antenna as described in the vertical section of the antenna radiation pattern in Figure 2.10. Tags have no mobility and they remain in the antenna's range during the whole scenario.

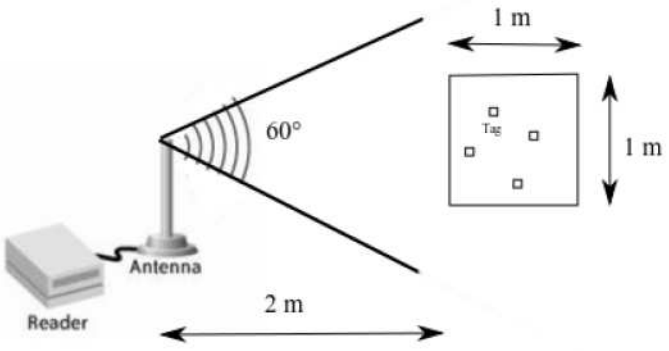


Figure 2.10: RFID fixed tags scenario schema

This scenario with one tag is run with the following command:

./waf –run rfid-fixed-tags

It is enriched with some options including the number of tags considered, as follows:

./waf –run "rfid-fixed-tags –tagsNumber=100"

In this scenario, results are printed on the terminal screen and composed of tag position, tag's EPC identifier, exchange duration and number of collisions detected. Note that, EPC identifiers are generated randomly, each tag generates its identifier in each inventory round.

### 2.6.2 RFID moving tags scenario

This scenario corresponds to reception/expedition control in the supply chain, and checkout control in a supermarket for example. This scenario differs from the previous one by adding mobility to tags and it is characterized by:

- An RFID reader that is permanently sending Query, QueryRep and QueryAdjust in order to identify tags getting into the reader's antenna range.
- An RFID tag population randomly distributed as in a pallet with a size of 1m x 1m x 1m, 1 meter far from reader's antenna. Tags start out of the antenna's range and they are moving with constant velocity of 1 m/s and they are getting into and out the antenna's range.

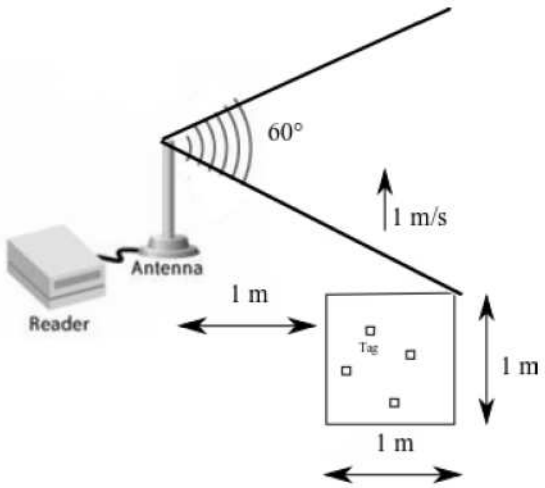


Figure 2.11: RFID moving tags schema

This scenario with one moving tag is run with the following command:

./waf –run rfid-mobile-tags

There are a number of command-line options available to control the default behavior. For instance, if we want to simulate a population of 100 tags, we specify the number of tags in

command line as follows:

./waf –run "rfid-mobile-tags –tagsNumber=100"

In addition to the results of previous scenario, the terminal prints the number of tags that the reader has been identified because some times the reader cannot identify all tags because of the very short crossing time of antenna's range.

### 2.6.3 RFID signal attenuation scenario

In this scenario we investigate the power received by tag during exchange operations. It is characterized by:

- An RFID reader that is permanently sending Query, QueryRep and QueryAdjust in order to identify a tag in the reader antenna's range.
- A static RFID tag.

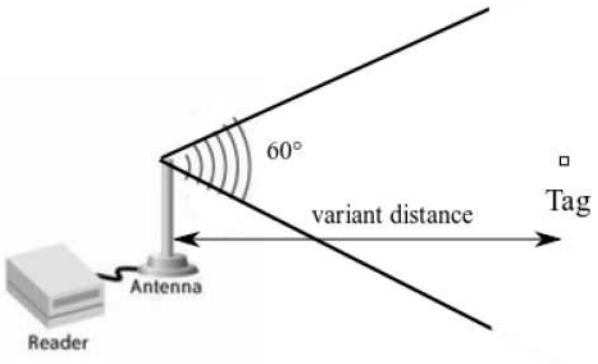


Figure 2.12: RFID signal attenuation schema

The following command is running the scenario:

./waf –run rfid-signal-attenuation

In this scenario, one tag is created in a fixed position and is identified, another one is created in a farther position and is identified until reaching the power tag receiver threshold. As a result, we get a graph that shows the power evolution according to the tag position. Scenario execution stops when power threshold is reached, so the tag could not receive messages from a farther position.

## 2.7 Simulation results and interpretations

In order to get significant results on the identification performances for different tags population i.e. identification time, number of collisions, and signal attenuation; each of the previous scenarios is executed one thousand times leading to average results.

### 2.7.1 RFID fixed tags simulation

The RFID fixed tag scenario (cf. section 2.6.1) is launched for different tag populations 1, 10, 100, 256, 500 and 1000. We get the identification time and number of collisions during the inventory as shown in table 2.4.

| Tag population | 1 | 10 | 100 | 256 | 500 | 1000 |
|---|---|---|---|---|---|---|
| Identification time (sec) | 0.009 | 0.038 | 0.350 | 0.875 | 1.690 | 2.924 |
| Number of collisions | 0 | 2.425 | 19.053 | 48.874 | 95.65 | 191.032 |

Table 2.4: Simulation results for the RFID fixed tag scenario

Note that, when we decrease the number of QueryRep (Figure 2.7), the identification time decreases rapidly for small tag populations, from 1.482 s to 75 ms for 100 tags, and from 2.174 s to 400 ms for 500 tags. However this time increases rapidly for large tag populations, from 2.9 s to 5 s for 1000 tags.

### 2.7.2 RFID moving tags simulation

The RFID moving tags scenario (cf. in section 2.6.2) is tested for different tag populations 1, 10, 100, 256, 500 and 1000. This results the identification time, the number of collisions and the number of tags that have been identified.

The maximum number of tags or the threshold that the reader can identify with the predefined velocity and distance from the reader is a very interesting result. For example when 1000 tags are configured to be identified, the reader could not identify all of them as demonstrated in table 2.5.

Note that, when we decrease the number of QueryRep (Figure 2.7), we obtain a similar observation as in the fixed tags scenario (cf. 2.7.1) but this time it is manifest in the reduced number of identified tags.

| Tag population | 1 | 10 | 100 | 256 | 500 | 1000 |
|---|---|---|---|---|---|---|
| Number of identified tags | 1 | 10 | 100 | 256 | 500 | 781,366 |
| Identification time (sec) | 0.40 | 1.293 | 1.482 | 1.629 | 2.174 | 2.980 |
| Number of collisions | 0 | 0,154 | 12,7 | 53,697 | 97,825 | 152,206 |

Table 2.5: Simulation results for the RFID moving tag scenario

### 2.7.3 RFID signal attenuation simulation

The RFID signal attenuation (cf. section 2.6.3) intends to study the evolution of the power received by the tag according to the distance between the tag and the reader. Figure 2.13 shows the power attenuation relatively to the distance that separates the tag from the reader.

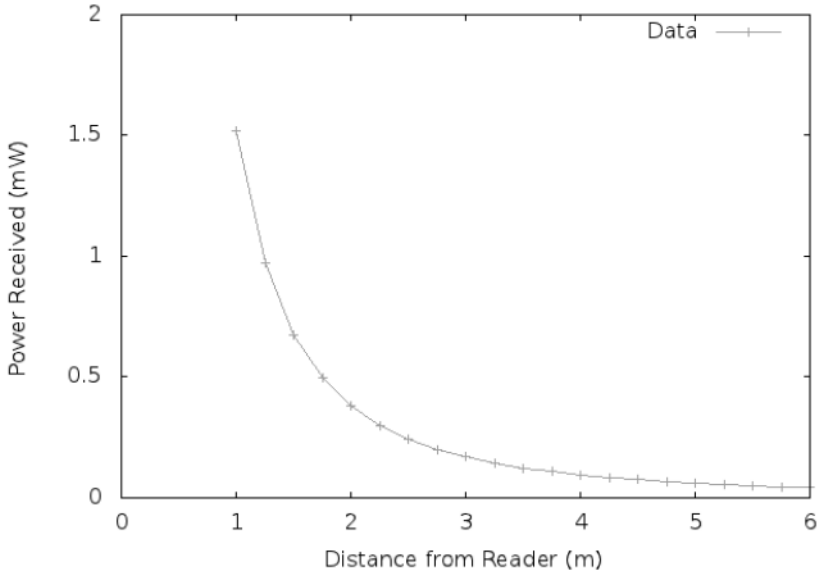


Figure 2.13: Power attenuation during inventory round

### 2.7.4 Interpretations

The identification in the RFID fixed tags scenario is very fast as demonstrated in Figure 2.14, around 9 ms per tag.

From the point of 10 tags, the identification time increases linearly: 38 ms for 10 tags, 350 ms for 100 tags, 875 ms for 256 tags and 1690 ms for 500 tags. For 1000 tags, the slope of the graph decreases slightly, it takes only 2.924 sec thanks to the choosing Q parameter algorithm which is more adapted for a large tag populations (Figure 2.1).

In the moving tags scenario, the identification is slower than in the fixed tags scenario. This is due to the mobility aspect of tags and has two causes. First, tags are at the beginning out

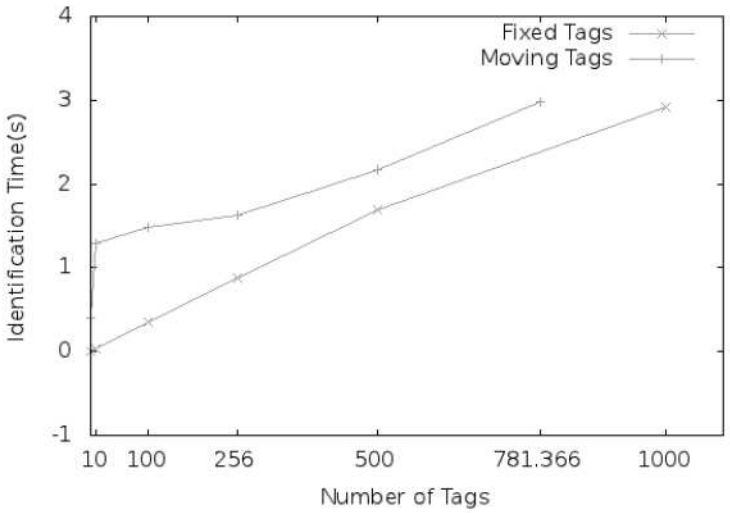


Figure 2.14: Identification time duration

of the antenna's range, so it takes time to get into the range. Second, when we have a large moving tag populations, the identification of tags is done group by group. Indeed, a group of tag populations in the same area is receiving commands Select and Query; in the same time, another group does not receive these commands because it is out of the antenna's range; when it comes into the antenna's range it does not respond to the QueryRep/QueryAdjust sent by the reader as it still waiting for commands Select and Query for a new inventory round which produces more collisions for a large tag population which consequently increases the identification time as we illustrate below in the analysis of collisions.

As deduced from figure 2.11, the pallet is fully inside the reader's range after 1.57 sec and is fully out of the range after 3.31 sec. The identification of one tag takes 0.40 sec as the duration depends on its position in the pallet. If at the head, it is identified quickly, if at the end, the reader is waiting until the whole container gets in to be identified. The identification is done smoothly for 10, 100, 256, and 500 tags, but the identification of 1000 tags fails due to the reader having not enough time to exchange information with all of them when they are moving at 1 m/s speed. Only 781 tags over 1000 are successfully identified.

Another important result about the identification quality is the number of collisions for each tag population. As shown in Figure 2.15, the collision rate is around 20 % of the number of tag populations configured initially in the two scenarios. For example, for 1000 tags, we get 191 collisions. Up to 100 tags, the number of collisions for mobile tags is lower than for fixed scenario. This drop can be explained by the manner that the reader identifies tags: when they are fixed, the reader tries to identify all of them at the same time (in one inventory round). But

when they are moving, they are identified in many inventory rounds i.e. group by group of tags as soon as they get into its range. Indeed, if the number of tags is not large enough, the first groups of tags are identified and when last groups come in the antenna's range the first groups are out of the antenna range because the width of antenna's range is smaller than the length of the pallet.

However, for more than 100 tags, we got the opposite observation. This is due to the large number of tags. A large group of tags at the head of the pallet is receiving Select and Query command (an inventory round) so collisions are still occurring when another large group is getting into the antenna's range, then tags of the previous group still in the antenna's range, and the entering groups are receiving Select and Query command to be identified (a new inventory round), so the number of collisions in the resulting group is very likely greater than that of the entering group only. If we continue this reasoning, we guess that the number of collisions and the identification time will likely be greater for moving tags scenario than in a fixed tags scenario for larger tags population.

Note that, when tags are moving, only about 781 tags are identified. This is why for 1000 tags the collision phenomenon is more significant than in fixed tags scenario (Figure 2.15).

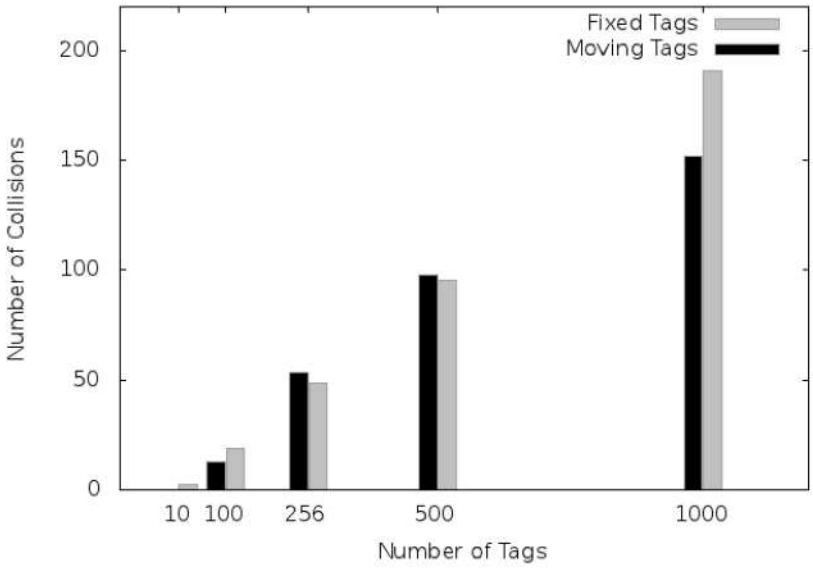


Figure 2.15: Collision detection

Figure 2.13 shows that the power level decreases rapidly when the distance between the tag and the reader is in the range of 1 to 3 meters. For higher distance, the power decreases more slowly.

A -14 dBm tag sensitivity is defined in the initial configuration and the model implemented in the channel layer allows the communication between the tag and reader up to 6 meters.

## 2.8 Conclusion

In this chapter, we described our ns-3 module to simulate the universal standard for low-cost RFID systems EPCGen2. It is the first simulator that can predict identification performances in this standard.

The main result is the high scalability of this standard i.e. the very short identification time for large tag populations (2.924 s for 1000 fixed tags), thus excluding any use of the symmetrical key cryptography in EPC Class-1 Generation-2 standard or in any future scalable standards for low-cost RFID systems.

This module is more realistic than existing simulators in the literature as it implements the most commonly used functionalities of EPCGen2 with its easy implement adaptation. This tool is very useful for tuning adequate of EPCGen2 system before real deployment, it can be used in many research challenges related to RFID technology such as the security, privacy, and how to speed up the identification of large tag populations. It will soon be available for free download.

This module can be further developed to simulate advanced scenarios and to reflect more realistic RFID systems. For example, a next step can be the design of algorithms for dynamic adjust of reader configuration in order to accelerate the identification, for example by estimating the number of tags to be identified.

In the next chapter, we focus on the scalability in low-cost RFID systems and we propose an adaptation of NTRU public cryptosystem to encrypt low data content and a scalable authentication protocol based on this adaptation.

# Chapter 3

# A scalable lattice-based authentication

In this chapter, we consider the scalability in low-cost RFID systems and we present an adaptation of NTRU public cryptosystem for constrained devices and new scalable asymmetrical authentication protocol based on this NTRU's adaptation specially designed for low-cost RFID. Thanks to properties of the polynomial ring in which NTRU operates we have ensured that the tag encrypts low data content using only addition and right circular shifts.
The proposed authentication protocol guarantees privacy, high scalability level and low implementation complexity. It takes advantages of NTRU and HMAC features, and is resistant to all the classical security attacks including replays, tracking, man in the middle attacks, etc.
This chapter is organized as follows. Section 3.1 introduces works related to RFID scalable authentication schemes. Section 3.2 gives a description of lattice and NTRU cryptosystem. Section 3.4 then describes our scalable asymmetrical mutual authentication protocol and Section 3.5 discusses its robustness to security and privacy threats. Performance issues aspects are also given in Section 3.6 before conclusions in Section 3.7.

## 3.1 Related Works

Contrary to secret key cryptography (symmetric) that requires the secret sharing between two communicating entities, public key cryptography permits two entities that have never met before to securely exchange information over an insecure channel, and to mutually authenticate themselves. Public key cryptography relies on two keys: a public key which is publicly known,

and a private key which is kept secret, hence the name asymmetrical key cryptography. The two keys are related by a mathematical equation that is difficult to solve without breaking a hard mathematical problem like the factorisation problem.

Many public key cryptography techniques have been proposed like RSA [47], Rabin [46], Elgamal [13], elliptic curve cryptography (ECC) [39, 34]. RSA, Elgamal and Rabin are widely used today, the ECC can achieve the same security level of RSA with a shorter key, which makes it suitable for small devices. The asymmetrical key cryptography is superior to secret key cryptosystems in key management and it is suitable for large scale deployment, distributed and open systems. However, it requires a higher computational overhead than symmetric key cryptography.

It has been demonstrated that it is possible to implement some public key cryptosystems, such as NTRU and ECC, on RFID tags [5, 7, 16]. However, the most used RFID tags are of very low-cost and they cannot support the standard public key cryptography because of its low capacity in terms of computation, memory, and power, which requires new asymmetrical approaches. This topic of research has not been much addressed by researchers. Only few ones have been proposed in the literature. Peeters et al. [43], Batina et al. [8], Kaya et al. [33], Lee et al. [35] propose RFID authentication protocols based on elliptic curve cryptography, but they require the tag to implement scalar-point multiplications in the tag which is still beyond current capabilities of low-cost RFID tags.

In Rabin [46] cryptosystem, the cipher text is the scare of the plain text calculated modulo a composite integer n. The modulo computation is more expensive than square computation or integer multiplication. Shamir proposes in [50] a randomized variant of Rabin scheme that does not require modulo computation, the idea is to add a multiple of the composite integer n to the scare of the plain text to obtain the cipher text. Oren et al. show that this randomized version can be implemented in less than 5000 GEs [42].

Another type of asymmetrical cryptography is the code-based cryptography which is very lightweight in term of computation and it can be efficiently implemented on hardware as it requires only simple bitwise operations. However, its shortcoming is the large size of keys, hundreds of megabits at least, which makes it beyond capabilities of low-cost RFID tags because the storage is the most expensive part in the hardware. In [49], *Sekino et al.* propose a code-based authentication scheme based on the Niederreiter public key cryptosystem [41]. The scheme reduces Niederreiter public key size, but it is still too large for low-cost tags.

## 3.2 Lattice and NTRU cryptosystem

### 3.2.1 Lattice theory

If $L$ is non-empty part of $\mathbb{R}^n$, so $L$ is a lattice if and only if $L$ is a discrete additive subgroup. Geometrically, a lattice is a repeating arrangement of points in Euclidean space. If $b_1$, $b_2$,..., $b_k$ are linearly independent vectors of $\mathbb{R}^n$, then a lattice $L$ generated by $b_1$, $b_2$,..., $b_k$ is the set:

$$L = \{\sum_{i=1}^{k} a_i b_i \mid \forall i \in \{1, ..., k\},\ a_i \in \mathbb{Z}\}$$

Figure 3.1 presents an example of 2 dimensional lattice.

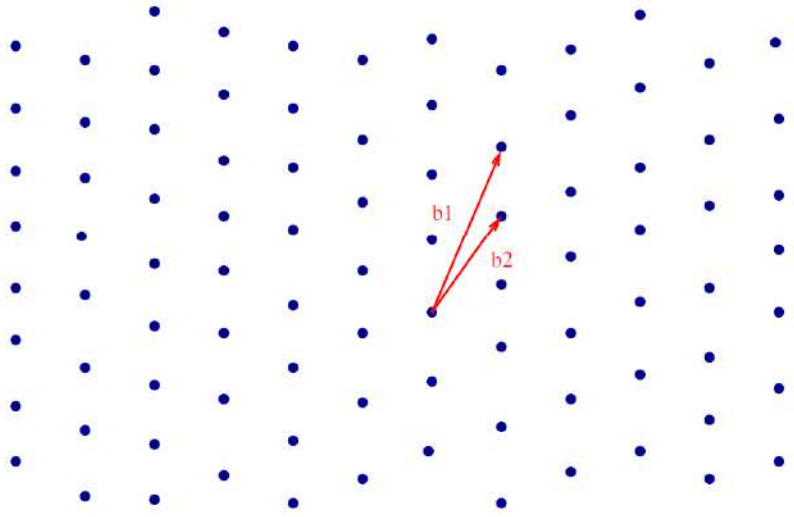


Figure 3.1: Example of 2 dimensional lattice

In a large dimension lattice, natural problems arise such as the search for the shortest vector, or the search for a the closest point to a given point.

**The Shortest Vector Problem (SVP)** Given a base B i.e. set of vectors linearly independent of a lattice $L$, find the smallest possible nonzero vector of $L$, i.e. find $v \neq 0$ such that $||v||$ is minimal. Where $||.||$ is the euclidean norm, if x=($x_1$,...,$x_n$), then $||x|| = \sqrt{< x, x >} = \sqrt{\sum_{i=1}^{n} x_i^2}$.

**The Closest Vector Problem (CVP)** Given a base B of a lattice $L$ and a vector $w \notin L$, find a vector $v \in L$ that is the closest to $w$, i.e. $||v - w||$ is minimal.

The SVP and CVP problems are known as NP-hard.

### 3.2.2 NTRU cryptosystem

NTRU is a probabilistic public key cryptosystem proposed by *Hoffstein et al.* [26], it is considered as secure by the standards IEEE 1363.1 [3] and X9 [4]. NTRU is one of the fastest public key cryptosystems, it is suitable for systems which can not be easily updated as designed for long-term data protection.

NTRU operations are performed in the polynomial ring $\Re = \mathbb{Z}[X]/(X^N - 1)$, where N is a positive prime, defining the dimension of the ring $\Re$. NTRU depends on three integer parameters (N, p, q) where $p$ and $q$ are relatively prime, and four sets $D_f$, $D_g$, $D_r$, $D_m$ of small polynomials of $\Re$.

#### 3.2.2.1 Key generation

One chooses randomly two polynomials $f$ and $g$ in $D_f$ and $D_g$, respectively; such that $f$ is invertible modulo $q$ and modulo $p$. Let $f_q$ be the inverse of $f$ modulo $q$ and $f_p$ the inverse of $f$ modulo $p$, the public key is $h = p{*}g{*}f_q \ (mod\ q)$, the corresponding private key is $(f,\ g)$.
Note that, generally the inverse of a small polynomial is a large polynomial i.e. with large coefficients, so the public key $h$ is a large polynomial.

#### 3.2.2.2 Encryption

Fast encryption of a message with NTRU includes three operations: Transform the message $m$ into a polynomial of $D_m$, randomly choose a polynomial $r \in D_r$ and calculate the cipher $e = r{*}h + m \ (mod\ q)$.

#### 3.2.2.3 Decryption

In knowing the value of $e$ in $\Re$ instead of $\Re \ mod\ q$ it is possible to obtain the message $m$ by applying a reduction $mod\ p$ because the public key $H$ is a multiple of $p$. However, the reduction $mod\ q$ on the cipher text $e = r{*}h + m \ (mod\ q)$ prevents a such reduction because $p$ and $q$ are relatively prime.
That is where the private key $(f,\ g)$ comes in : calculate $a = f{*}e \ (mod\ q) = p{*}r{*}g + m{*}f \ mod\ q$, where $p{*}r{*}g + m{*}f$ is small by construction, this means that we can obtain $p{*}r{*}g + m{*}f$ in $\Re$. So now we can apply the reduction $mod\ p$ on $p{*}r{*}g + m{*}f$ to obtain $m{*}f \ mod\ p$ and then the message $m$ because $f$ is invertible modulo p.

## 3.3 From a lattice point of view

Key generation, encryption, and decryption process of NTRU can be seen from a lattice perspective.

The NTRU lattice is $L_{NTRU} = \{(u, v) \in \Re^2 \mid u*h - v = 0 \ (mod\ q)\}$, this lattice is created from the public key $h$ such that the private key $(f,\ g)$ is in this lattice. Indeed, if we suppose that the multiplication by $p$ is done during the encryption process i.e. $h = g*f_q \ (mod\ q)$, then $f*h - g = 0 \ (mod\ q)$. This means that $(f,\ g)$ is in the lattice $L_{NTRU}$. On the other hand, if all the coefficients of $f$ and $g$ are in $\{-1,\ 0,\ 1\}$, it is likely that $(f,\ g)$ is the shortest vector of $L_{NTRU}$. In finding the shortest vector of $L_{NTRU}$, it is certain to find the private key $(f,\ g)$.

The encryption process is to create a point which is not on the lattice but close to a lattice point: the lattice point $(p*r,\ p*r*h\ mod\ q)$ is slightly offset by adding $(0,\ m)$ to obtain a $(p*r,\ e = p*r*h + m \ (mod\ q))$ which is not in the lattice. So, geometrically NTRU decryption corresponding to a CVP over NTRU lattice.

## 3.4 The proposed protocol

Our protocol is an asymmetrical probabilistic mutual authentication based on the NTRU public key cryptosystem [26]. *Atici et al.* have designed in [5] an NTRU's architecture for encryption/decryption that requires 10.8 *kgate* for $(N, p, q) = (167, 3, 128)$, our solution does not dedicate such amount of resources for the classical operations of NTRU on the tag, because all complex operations of NTRU such as modular arithmetic, polynomials multiplication are done at the server while the tag implements only lightweight operations.
The proposed protocol is specially designed to simultaneously support privacy (tracking resistance), high scalability level, and security protection at a moderate operational overhead.

### 3.4.1 Our adaptation of NTRU to low-cost RFID tags

The public key cryptosystem NTRU handles polynomials in the ring $\Re = \mathbb{Z}[X]/(X^N - 1)$ consequently, all the polynomials are of degree less than N.
Any element $f$ of $\Re$ is represented by:

$$f = (f_0, f_1, ..., f_{N-1}) = \sum_{i=0}^{N-1} f_i x^i$$

Multiplication in the ring $\Re$ is a convolution $mod\ X^N - 1$. In $\Re$, the addition of two polynomials is done term by term. However, the ring $\Re$ is cyclic, this means that a multiplication by $X$ is equivalent to rotate the coefficients: the coefficient of $X^i$ becomes the coefficient of $X^{i+1}$ (and since multiplications are $mod\ X^N - 1$, the monomial $X^N$ is equal to 1). Specifically, this convolution works as follows:

Let's $(f, g) \in \Re^2$ the convolution product defined by:

$$(f*g)(X) = \sum_{i=0}^{N-1} h_i X^i,\ \forall\ 0 \leq k \leq N-1,\ h_k = \sum_{i+j \equiv k\ mod\ N}^{k} f_i g_j$$

If $f'$ is a right circular shift of $f$ by i-position, the product $f'*g$ is the right circular shift of $f*g$ by i-position. Indeed, a right circular shift of $f$ by $i$-position is exactly equal to $rot_i(f) = X^i*f\ mod\ (X^N - 1)$ where $X^N*f\ mod\ (X^N - 1) = f$.

So $f'*g = X^i*f*g\ mod\ (X^N - 1) = X^i*(f*g)\ mod\ (X^N - 1) = rot_i(f*g)$ , then:

$$rot_i(f*g) = rot_i(f)*g \qquad (1)$$

If we add $f*g$ to its right-shift rotation by $i$-position (1), we obtain: $f*g + rot_i(f*g) = f*g + X^i*f*g\ mod\ (X^N - 1) = (f + X^i*f)*g\ mod\ (X^N - 1) = (f + rot_i(f))*g$, then:

$$f*g + rot_i(f*g) = (f + rot_i(f))*g \qquad (2)$$

Each coefficient of a polynomial in $\Re\ mod\ q$ is less than $q$, so it can be written on $\lceil log_2(q-1) \rceil$ bits. Then, each rotation of $r*h\ mod\ q$ by $s - bit$, where $s$ is a multiple of $\lceil log_2(q-1) \rceil$, corresponds to the new product $r'*h\ mod\ q$, where $r'$ is a right circular shift of $r$ by $s - bit$ (cf. Equation (1)), and each $r*h + rot_s(r*h)\ mod\ q$ corresponds to another product $r_s*h\ mod\ q$ where $r_s = r + r'$ (cf. Equation (2)). So, if the tag has in memory $r*h\ (mod\ q)$, it can construct easily an $r_s*h$ and encrypts a challenge $m$ by adding it to $r_s*h$. This point will be later used in this chapter.

Note that, in $NTRU$ specifications, $r$ is any small polynomial, its coefficients can be chosen in {-1, 0, 1} or {0, 1} to simplify the implementation or to attain very high security level according to the implementation standard [1], [27].

In our approach, few coefficients of this polynomial ($r_s$) can be equal to 2 or $-2$, such modification does not have a significant impact on the security of $NTRU$. Indeed, breaking the private key is an SVP problem in the NTRU's lattice $L_{NTRU} = \{(u, v) \in \Re^2 \mid u*h - v = 0\ (mod\ q)\}$ which is independent of the choice of $r$. On the other hand, the decryption problem can be seen as a CVP problem: the cipher of a message $m$ is $e = r*h + m\ (mod\ q)$; this means that the

vector $(0, e)$ is close to the vector $(pr, pr*h \ mod \ q)$ of the NTRU's lattice. More precisely, the difference between the two vectors is $(pr, m)$ which is by definition very short. Consequently, if few coefficients of $r$ are equal to 2 and/or $-2$, the difference $(pr, m)$ between these two vectors remains very short. Then, for an attacker, decrypting a ciphertext with no knowledge of the private key is still a difficult CVP problem in the NTRU's lattice.

| | |
|---|---|
| $r*h \ (mod \ q)$ | Designed by $r*h$ |
| $r^{'}*h \ (mod \ q)$ | Designed by $r^{'}*h$ |
| $\oplus$ | Exclusive-or operator |
| $HW(x)$ | Hamming weight of $x$ |
| $rot(x, y)$ | Right circular shift over $x$ by $HW(y)$ |
| $rot^{-1}(x, y)$ | Left circular shift over $x$ by $HW(y)$ |
| $rot(x, K'y)$ | Right circular shift over $x$ by $K$ times the $HW(y)$ |
| $\lceil x \rceil$ | Smallest integer not less than $x$ |
| $H_{k_t}()$ | Hash-based Message Authentication Code (HMAC) |
| $k_t$ | long-term secret key |
| $x_{[0,s-1]}$ | the $s$-least significant bits of $x$ |

Table 3.1: Notations.

### 3.4.2 Initialization

Each tag is initialized with a random long-term secret key $k_t$ and two secret polynomials (binary vector): an identity $id \in D_m$ that is unique at the back-end, and a polynomial $r*h$ calculated modulo q in $\Re$ where $r$ is randomly generated in $D_r$. The back-end stores only one private key $(g, \ f)$ to authenticate all tags, and an $id$ for each tag.

### 3.4.3 Description

As described in figure 3.2, based on the notations given in Table 3.1, when the tag is queried, it reads the pre-computed value $M$, it calculates $M = H_{k_t}(M)$, $r_t = M_{[0,s-1]}$ and it constructs a $n$-bit binary vector $m = B2P(M)$, where $B2P : \{0,1\}^N \rightarrow D_m$ converts a sequence of bits into a binary polynomial where each coefficient is written in $n = \lceil log_2(2q-1) \rceil$, and $n = \lceil log_2(2q-1) \rceil N$.

Note that, $m \in D_m$ as each of its coefficients is in $\{0, 1\}$. Then the tag reads the pre-computed value $r*h$ and calculates $r_s*h = r*h + rot(r*h, \lceil log_2(q-1) \rceil' r_t)$ and replaces $r*h$ and $M$ with $rot(r*h, \lceil log_2(q-1) \rceil' r_t)$ and $H_{k_t}(M)$ respectively. Note that, each coefficient of $r_s*h$ is less than $2q-1$, this is why each coefficient of $m$ and $r_s*h$ is written in $\lceil log_2(2q-1) \rceil$ bits.

After that, the tag calculates and sends back to reader $e_1 = r_s*h+m$. Note that, if the previous authentication attempt has been successfully achieved, in its polynomial form, $e_1$ can be written as $e_1 = r_s*h + m = (X^{HW(r_t)} + 1)*r*h + m \ mod(X^N - 1)$ (cf. section 3.4.1).

If there is one or more failed previous authentication attempts, in its polynomial form, $e_1$ can be written:

$$e_1 = (X^i + X^j) * r * h + m \ mod(X^N - 1), (i, j) \in \mathbb{N}^2 \qquad (3)$$

As such, the tag only does two sum operations of four binary values, few xor operations, some right circular shifts, one HMAC.

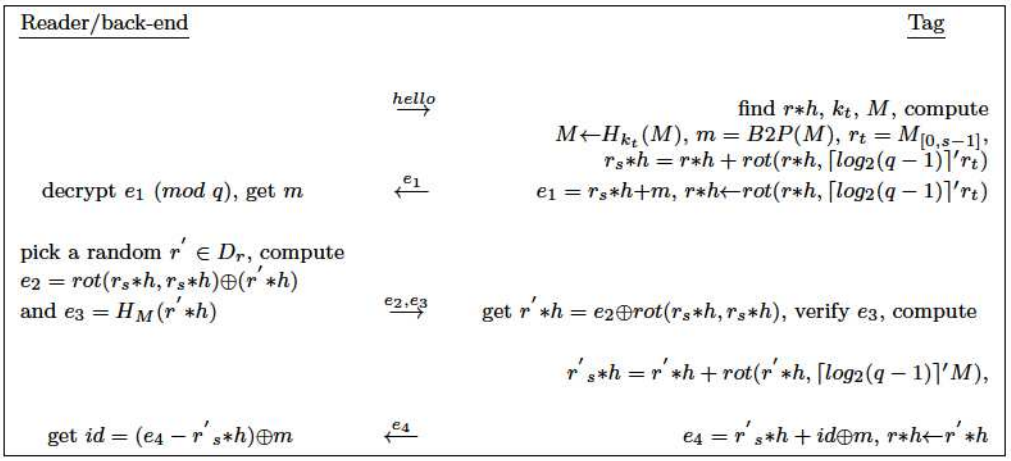


Figure 3.2: The proposed protocol for $p = 2$

Upon receiving the tag's response ($e_1$), the reader/back-end calculates and decrypts $e_1$ $(mod\ q)$ to retrieve $m$. Note that, $e_1$ $(mod\ q)$ is a valid NTRU ciphertext.

That is, $e_1$ $(mod\ q)$ is computed by the reader as each coefficient of $e_1$ is at most equal to $q-1$: each coefficient of $m$ is in $\{0, 1\}$ and each coefficient of $r_s*h$ used in the computing of $e_1$ is between 0 and $q-1$ (because each coefficient of $r*h$ is between 0 and q-1).

The reader/back-end decrypts $e_1$ $(mod\ q)$ using the secret key $(g,\ f)$ to retrieve $m$. It then generates randomly a polynomial ($n-$bit pseudo-random sequence) $r' \in D_r$ and computes $r'*h$ $(mod\ q)$ in $\Re$, $e_2 = rot(r_s*h, r_s*h)\oplus(r'*h)$, and $e_3 = H_M(r'*h)$. Then the reader/back-end sends these values to the tag. Upon receiving $e_2$, the tag retrieves $r'*h$ as $r'*h = e_2\oplus rot(r_s*h, r_s*h)$, it authenticates the back-end/reader by checking the correctness of $e_3$, which authenticates $r'*h$, it can be generated only by the legitimate reader thanks to HMAC. If $e_3$ is correct the reader/back-end is authenticated. Otherwise the tag aborts the session concluding that the reader failed to decrypt $e_1$ correctly.

If reader/back-end is authenticated, the tag sends back to reader $e_4 = r'_s*h + id\oplus m$, where $r'_s*h = r'*h + rot(r'*h, \lceil log_2(q-1)\rceil' M)$, and replaces $r*h$ with $r'*h$. Note that, similarly to Equation (3) and from the description in section 3.4.1 , in its polynomial form, $e_4$ can be written:

$$e_4 = (X^{HW(M)} + 1) * r' * h + m \oplus id \ mod(X^N - 1) \quad (4)$$

Let $id_{temp}$ the binary polynomial defined by $id_{temp} = m \oplus id$, this means that $id_{temp_i} = m_i \oplus id_i$, where $0 \leq i \leq N - 1$.

The tag's next authentication session is executed with the new value $r' * h$.

To counteract replay attacks, the reader retrieves the tag's identity by $id = (e_4 - r'_s * h) \oplus m$ and not by the decryption of $e_4 \ mod \ q$ using the private key $(f, \ g)$ which allows reader to personalize each session.

Note that the reader/back-end can authenticate a tag without knowledge of the tag's identity. This feature is very much interesting in some applications, specially in supply chains. Off-line authentication of a tag (with no exchanges with a central database) is made possible as the same private key $(f, \ g)$ is used for every tag.

### 3.4.4 Roaming support

Inter-domain authentication of tags, especially in the supply chain environment is not well investigated in the literature.

Our proposed protocol can be extended to securely support the inter-domain tag authentication. Just before moving to another domain, the last authentication to the domain leads to the tag being updated with the next domain public key $h'$.

This ownership transfer is very simple but it requires a minimum level of mutual reliance between supply chain partners. Schemes that are specially designed to ownership transfer have been proposed in [14, 37, 40].

## 3.5 Security and privacy analysis

The resistance of our protocol against the classical security attacks is directly derived from the properties of NTRU and HMAC.

In the following analysis, we only consider vulnerabilities over the reader-tag channel. The channels between the reader and the back-end are considered as safe as both equipments have computing and battery resources, they are under the same administrative domain and they can implement any security protocols.

The attacker is assumed behaving according to the Dolev-Yao model, i.e. having full control over the wireless channel to replay, modify and store exchanged messages (cf. Section 1.2).

### 3.5.1 Resistance to replay attaks

For each authentication session, tag and reader/back-end generates new pseudo-random values $m$, $r_s{*}h$ and $r^{'}{*}h$, thus making messages randomized (personalized) for each session.

A replay attack to the reader is unlikely to occur, as the reader/back-end is assumed implementing a good pseudo-random polynomial generator, so they are unlikely to generate the same $r^{'}{*}h$.

A replay attack to the tag can only be successful if the attacker prevents the tag from receiving ($e_2$, $e_3$) (e.g. by transmitting a jamming signal) and the tag generates the same sequences $m$ and $r_s{*}h$ in the next authentication attempt.

The probability this attack is successful is negligible thanks to HMAC and the construction method of $r_s{*}h$, however, as discussed in the desynchronization attack, in case of success, it does not lead to any desynchronization between tag and reader/back-end.

### 3.5.2 Resistance to man in the middle attacks

The aim of the attacker is to be authenticated as a legitimate tag or reader. We have demonstrated that, the attacker can not break NTRU using the modification that we have introduced on the choice of $r$ (cf. section 3.4.1).

Suppose that the attacker is passive and he wants to be authenticated as a legitimate tag, he has to produce a valid message $e_4$ or to retrieve the tag secret identity $id$.

It's clear that the attacker cannot produce a valid $e_4$ from the previously exchanged messages between the tag and readers because any modification in a previous tag's response $e_1$ will be detected in the attacker's response $e_4$ thanks to the random value $r^{'}{*}h$ generated by the reader itself. On the other hand, the attacker cannot retrieve the identity of the tag because he cannot break NTRU.

If the attacker is active, he will try to take advantage of the fact that the number of $r_s{*}h$ between two legitimate authentication sessions is limited.

To retrieve the identity of the tag $id$, he must first find the pseudo random value $m$ or $r{*}h$ (initial values assigned to the tag during the previous mutual authentication session). For this, he queries the tag between two legitimate mutual authentication sessions. He has to retrieve the polynomial $m^{(k)} - m^{(i)}$ for a specific $k$ and several $i$ in order to find $m^{(k)}$ and $r{*}h$, where $m^{(j)}$ corresponds to the $m$ of the j-th authentication attempt (attacker's query). Indeed, each $m^{(j)}$ is small polynomial with coefficients in $\{0, ..., p-1\}$, so if $p = 2$, $m^{(j)}$ is a binary polynomial, this

means that if the attacker finds a specific $k$ and some $m^{(k)} - m^{(i)}$ for several $i$, he can deduce some coefficients of $m^{(k)}$ when $m_l^{(k)}$ and $m_l^{(i)}$, $l = 1, ..., N-1$ are different and he can proceeds by brute force attack to get the rest of coefficients of $m^{(k)}$, then he gets ${r_s}^{(k)} * h$, $m^{(k+1)}$, etc. and proceeds by another attack to deduce $r*h$ because from the Equation (3) it's clear that:

$$r*h = \frac{{e_1}^{(k)} - m^k}{X^i + X^j} \ mod\ X^N - 1,\ (i,j) \in [\![0,\ \text{N-1}\ ]\!]^2$$

Note that, we can suppose that $(i,j) \in [\![0,\ \text{N-1}\ ]\!]^2$ thanks to calculation with $mod(X^N - 1)$.

However, this attack cannot be performed as the attacker cannot retrieve for a specific $k$ and several $i$, $m^{(k)} - m^{(i)}$. Indeed, suppose that the attacker begins his attack as described above, the tag responds to the i-th and to (i+1)-th attacker queries with ${e_1}^{(i)} = X^{k_i} * r*h + X^{k_{i-1}} * r*h + m^{(i)}\ mod(X^N - 1)$, and ${e_1}^{(i+1)} = X^{k_{i+1}} * r*h + X^{k_i} * r*h + m^{(i+1)}\ mod(X^N - 1)$, respectively (cf. Equation (3)), where $X^N * r*h\ mod(X^N - 1) = r*h$, and $(k_{i-1}, k_i, k_{i+1}) \in \mathbb{N}^3$. So,

$${e_1}^{(i+1)} - {e_1}^{(i)} = (X^{k_{i+1}} - X^{k_{i-1}}) * r*h + m^{(i+1)} - m^{(i)}\ mod(X^N - 1) \qquad (5)$$

Suppose that the attacker wants to obtain the value of $m^{(i)}$ from $e^{(i-1)}$, $e^{(i)}$, and $e^{(i+1)}$. Thanks to calculation with $mod(X^N - 1)$ and to simplify the reasoning we can write $k_i = \sum_{j=1}^{i} HW({r_t}^{(j)})$, then $k_{i+1} - k_{i-1} = (HW({r_t}^{(i+1)}) + HW({r_t}^{(i)}))$. To retrieve $m^{(i+1)} - m^{(i)}$ the attacker should eliminate the term $(X^{k_{i+1}} - X^{k_{i-1}}) * r*h$ in the Equation (5), this is possible only if $k_{i+1}$ equal to $k_{i-1}$ or if each of $k_{i+1}$ and $k_{i-1}$ is a multiple of $N$. However, this is unlikely to occur as $k_{i+1} = k_{i-1}$ means that the tag has generated successively two null values of $r_t$ but this scenario is unlikely to occur because HMAC generates good pseudo random sequences. On the other hand, the value of $s$ is chosen in such a way that $k_{i+1}$ and $k_{i-1}$ are never a multiple of $N$ in the same time, in other words $k_{i+1} - k_{i-1} = (HW({r_t}^{(i+1)}) + HW({r_t}^{(i)}))$ is less than $N$, so $s$ can be chosen such that its length is less than $N/2$. Then the attacker cannot retrieve $m^{(i+1)} - m^{(i)}$ and $m^{(i)} - m^{(i-1)}$ to try the attack described above on $m^{(i)}$.

However, let's $w$ the number of all possible values of an ${r_s} * h$ derived from one $r*h$ (initially assigned to the tag by the reader/back-end) between two legitimate authentication sessions, it's clear that $w = N + N - 1 + N - 2 + ... + N - (N-1)$. Then, $w = N^2 - \sum_{i=1}^{N-1} i = 1/2(N^2 + N)$, for $N = 251$, $w = 31626$. Consequently, if the attacker queries the tag more than $w$ times, there will certainly be ${e_1}^{(u_1)}$ and ${e_1}^{(v_1)}$ using the same ${r_s} * h$, so from the Equation (3) we deduce:

$${e_1}^{(u_1)} - {e_1}^{(v_1)} = m^{(u_1)} - m^{(v_1)}, (u_1, v_1) \in [\![1,\ \text{w}\ ]\!]^2 \qquad (6)$$

Note that, the attacker cannot accurately determine ${e_1}^{(u_1)}$ and ${e_1}^{(v_1)}$. If the attacker continues to query the tag (before the next mutual authentication), there will be ${e_1}^{(u_2)}$ and ${e_1}^{(v_2)}$ such

that ${e_1}^{(u_2)} - {e_1}^{(v_2)} = m^{(u_2)} - m^{(v_2)}$. However, and thanks to HMAC it is likely that each of $m^{(u_1)}, m^{(u_2)}, m^{(v_1)}$ and $m^{(v_2)}$ is different from each other. If we continue this reasoning we note that the attacker is not able to find in reasonable period of time for a specific $k$, several $i$ such that ${e_1}^{(k)} - {e_1}^{(i)} = m^{(k)} - m^{(i)}\ mod(X^N - 1)$. As such the attacker cannot retrieve either $m$ or $r*h$.

The attacker cannot proceed by the previously described attack (cf. Equation (6)) on the message $e_4$ in order to retrieve $id \oplus m$ or $r^{'}*h$ because $r^{'}*h$ changes for each $e_4$. However, the tag next authentication session will be executed with $r^{'}*h$ which has been generated by the reader in the previous session (session $k$), so once the session $k$ is successfully completed and before the next mutual authentication session happens, the attacker can start to query the tag. If he queries the tag more than $1/2(N + N^2)$ there *may be* ${e_1}^{(l)}$ such that ${e_1}^{(l)} = (X^{HW(M^{(k)})} + 1)*r^{'}*h + m^{(l)}\ mod(X^N - 1)$ (cf. Equation (4)), then: ${e_1}^{(l)} - {e_4}^{(k)} = m^{(l)} - m^{(k)} \oplus id, (k, l) \in \mathbb{N}^2$ (cf. Equations (6)). Even if he finds the right ${e_1}^{(l)}$, it's clear that he cannot retrieve the identity of the tag $id$ from this equation (he does not know either $m^{(k)}$ or $m^{(l)}$).

A means of increasing the number of $r_s*h$ that the tag can generate between mutual authentication sessions, is to store several values $r*h$ on the tag in such away that the tag uses randomly one of them in each authentication attempt.

On the other hand, the identity of the tag is encrypted (cf. Equation (4)) in such a way that the multiple transmission attacks on NTRU is avoided. Indeed, if the same message $m$ is transmitted many times using the same public key $h$, but with different random $r$'s, the attacker will be able to retrieve a large part of the message NTRU[26]. To counteract this attack, we encrypt in $e_4$, $id \oplus m$ instead of the static $id$ value, as $id \oplus m$ is different from one authentication session to another. As such, our protocol is resistant to man in the middle attacks.

### 3.5.3 Tag anonymity and resistance to tracking

The proposed protocol supports a sufficient level of privacy for low-cost RFID systems. Two scenarios arise according to whether the attacker is passive or active.

If the attacker is passive, i.e. he can not query the tag but he can eavesdrop communications between the tag and legitimate readers. He cannot track the tag as the messages exchanged over the wireless channel between the tag and the reader are randomized in each authentication session thanks to $m$, $r^{'}*h$ (cf. Equation (4)).

If the attacker is active, i.e. he can query the tag. The privacy is also guaranteed as the tag respond $e_1 = r_s*h+m$ is randomized in each attacker's request thanks to the construction method of $r_s*h$ and HMAC. Indeed, the attacker cannot track the tag between two mutual authentication sessions by trying to distinguish the specific $r*h$ used by the tag because in ${e_1}^{(i)} = (X^{k_i} + X^{k_{i-1}})*r*h + m^{(i)}\ mod(X^N - 1),\ (i,j)\ \in\ \mathbb{N}^2$ (cf. Equation (3)) the attacker cannot obtain information about specific coefficients of $r*h$ that allows him to track the tag.

However, this protocol does not support a high privacy level because the number of $r_s*h$ between two mutual authentication sessions is limited. One solution is to store several values $r_s*h$ on the tag, so that the tag use one of them in each authentication attempt.

### 3.5.4 Resistance to desynchronization attacks

The attacker can try to make the tag and reader/back-end out of synchronization by acting as an active man in the middle attacker, modifying $e_2$ in order to provide to the tag an invalid value of $r^{'}*h$, this modification will be necessarily detected in the HMAC value $e_3 = H_M(r^{'}*h)$ that guarantees the authenticity of $r^{'}*h$ because $M$ know only to the tag and the reader which has the private key $(g,\ f)$ that allows decryption of $e_1$ (the cipher value of $M$ (or $m$)). Indeed, suppose that, the attacker has modified the bit at index $k$ of $e_2$, with $e_{20}$ being the least significant bit of $e_2$. He cannot produce a valid value of $e_3$ because he does not know the values of $M$ (or $m$) and $r^{'}*h$ that must be used in this HMAC calculation. Then tag-reader/back-end synchronisation is guaranteed.

## 3.6 Performance evaluation

The proposed protocol is designed for low-cost RFID tags. All the complex operations such as polynomial multiplications over ring, random polynomial generation in $D_r$, computation of modulo, etc. are done at the server.

Let $n = \lceil log_2(q-1) \rceil N$, the tag only requires to store about $2N + n + 128$ bits and to support lightweight hash function like PHOTON [25] or KECCAK [32].

The computations by the tag are limited to two HMAC values, and an average of about $\lceil log_2(q-1) \rceil (N + S/2)$ bitewise right circular shift, addition of four polynomials, and $n + N$ bitwise xor operations. On the other hand, the reader/back-end implements right circular shift operator, and two NTRU algorithms to perform one encryption and one decryption each one with

a complexity of only $O(N^2)$. Moreover, the reader/back-end does not require any exhaustive search in keys to authenticate a tag.

## 3.7 Conclusions

In this chapter, we presented an adaptation of NTRU for low-cost RFID tag, and a lightweight mutual authentication protocol based on this adaptation.

This solution satisfies the security and privacy requirements for RFID systems. It benefits from the NTRU features like high security level, and fast encryption and decryption. It provides remarkable properties such as strong scalability and untraceability, and resistance to known classical security attacks.

The proposed solution can be implemented efficiently into low-cost tags, as tags are only required to implement a lightweight hash function, and bit-wise operations in $GF(2)$.

In the next chapter, we focus on zero-knowledge identification in low-cost RFID systems, and we present a coupon-based zero-knowledge identification protocol based on GPS and randomized GPS identification schemes.

# Chapter 4

# Zero-knowledge identification

In this chapter, we consider the zero-knowledge identification in low-cost RFID systems i.e. when the tag (prover) convinces the reader (verifier), it holds the secret information without revealing any information thereon. We propose two zero-knowledge approaches based on GPS and randomized GPS. The proposed approaches consist to store in the back-end precomputed values in the form of coupons.

The ideas are twofold. First, the coupons are stored only on the back-end and not on the tag, so the GPS-based variant protocol can be private, the number of coupons can be much higher than in other approaches thus leading to higher resistance to denial of service attacks for still low-cost tags, and consumed coupons can be easily replaced with new ones. Second, for authenticating to the reader, the tag only needs simple integer operations: one or two pseudo random number generations and two or three simple integer operations according to the variant (GPS or randomized GPS). As such, the implementation can be done in less than 1000 gate equivalents (GEs).

The computation and storage capacity required on the server remains reasonable. This approach is demonstrated to be robust against the classical security and privacy attacks performed over the wireless channel.

The proposed approach takes advantages of the GPS scheme, which is resistant to the classical security attacks including replays, tracking, man in the middle attacks, etc.

We first introduce works related to zero-knowledge identification in RFID, GPS and randomized GPS schemes in Section 4.1. Then we describe in Section 4.2 the proposed approaches and Section 4.3 discusses their robustness to security and privacy threats. Performance issues aspects are also given in Section 4.5 before conclusions in Section 4.6.

## 4.1 Related Works

### 4.1.1 Zero-knowledge identification

The most widely used authentication process today is based on a secret sharing between the entity to be authenticated (the prover) and the verifier entity as demonstrated in Figure 1.2. The identity is proved by the secret aspect of information making it critical, because if a third entity successfully gets the secret information, it can impersonate the legitimate prover.
The confidentiality of secret information can be relatively well preserved in a restricted (closed) environment. However, it becomes much more problematic in open systems such as some RFID applications. In addition, there is no guarantee that the verifier is not malicious e.g. when an RFID tag moves among many users who cannot be identified ahead of time.
These issues of conventional authentication schemes have lead to zero-knowledge schemes which are special cases of interactive proofs in which an entity, the prover must convince the verifier, it holds the secret information without revealing any information thereon.

A zero-knowledge identification scheme must satisfy three fundamental properties:

- Completeness: if the prover and the verifier are honest then the verifier always accepts the proof.
- Soundness: if the proof is false, the probability of cheating an honest verifier is very low.
- Zero-knowledge: the verifier is not able to draw any information about the secret of the prover.

A zero-knowledge identification scheme is generally three passes and one or more rounds. This means that in each round, three messages are transmitted between the prover and the verifier; commitment, challenge, and response as illustrated in Figure 4.1.

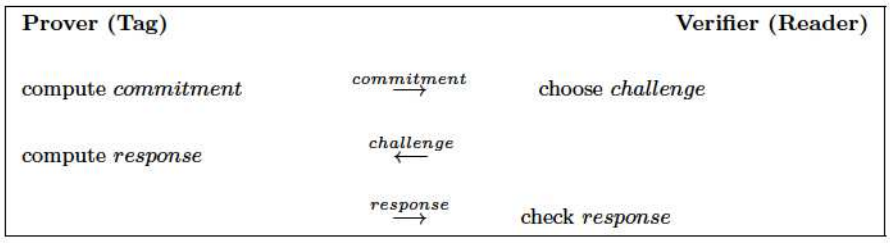


Figure 4.1: Zero-knowledge identification scheme

Many zero-knowledge identification schemes have been proposed such as Fiat-Shamir [17], Guillou-Quisquater [24], Stern [51], Schnorr [48], and GPS [21] protocols.

The best-known and most efficient ones in terms of transmission are Guillou-Quisquater and Schnorr schemes which are based on the difficulty of factoring and the discrete logarithm problem; respectively.

The best example to illustrate the fundamental ideas of zero-knowledge proofs was published by Quisquater et al. in their paper entitled "How to explain zero-knowledge protocol to your children" [45] which provides very simple explanations based on the story of "Ali Baba and the Forty Thieves".

As illustrated in Figure 4.2, the forked cave of Ali Baba connected at both ends A and B by a secret passage that can not be used without magic word.

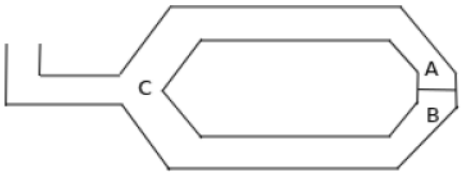


Figure 4.2: The cave scheme

The prover would like to convince the verifier that he knows the magic word, but he does not want to reveal the magic word to the verifier. To do this, they execute the following protocol:

- The prover gets into the cave and takes place at the extremity either A or B, for example by flipping a coin or rolling a die. The verifier is not allowed to see which passage the verifier takes.
- The verifier gets position in the cave at C and demands to the prover to return to C from the passage A or B chosen at random.
- The prover comes to C, if he can, along the desired passage.

The chance of success of a malicious prover to come from the desired passage is 1/2.

If the prover comes from the wrong passage, the verifier concludes that the prover is malicious. Otherwise, he starts to believe him. He can then declare himself satisfied and stop or continue to strengthen his faith in the prover.

By repeating this protocol forty times, the chance of success of a malicious prover is only one chance in a million million.

So, if the prover knows the magic word, he come through the passage (A or B) chosen by the verifier.

### 4.1.2 Lightweight zero-knowledge identification for RFID

The non secret sharing between the prover and the verifier in zero-knowledge schemes is very important in many RFID applications where the interrogator of an RFID tag cannot be identified ahead of time for example due to the mobility aspects of RFID tags like in a supply chain. In other words, there is no guarantee that the tag's interrogator is not malicious. However, zero-knowledge identification schemes requirements in terms of number of rounds and/or resources (computation and energy) are beyond capabilities of low-cost RFID tags.

Lightweight zero-knowledge approaches have not been much addressed by researchers, only few approaches have been proposed in the literature.

To the best of our knowledge one of the most suitable zero-knowledge schemes for RFID tags is the storage-computation trade-off approach of the famous GPS scheme [21] proposed by Girault [19], but the required storage capacity on the tag for a moderate security level is still beyond current capabilities of low-cost RFID tags.

The GPS scheme [21] named after its authors Girault, Poupard and Stern, is a lightweight variant of Schnorr scheme which requires one scalar-point multiplication (elliptic curve variant) and modular reduction in prover side, while GPS requires only one scalar-point multiplication. The elliptic curve variant of GPS scheme is illustrated in Figure 4.3.

GPS is now standardized by the international standard ISO/IEC 9798-5 [30], and it is listed in the final NESSIE portfolio [28].

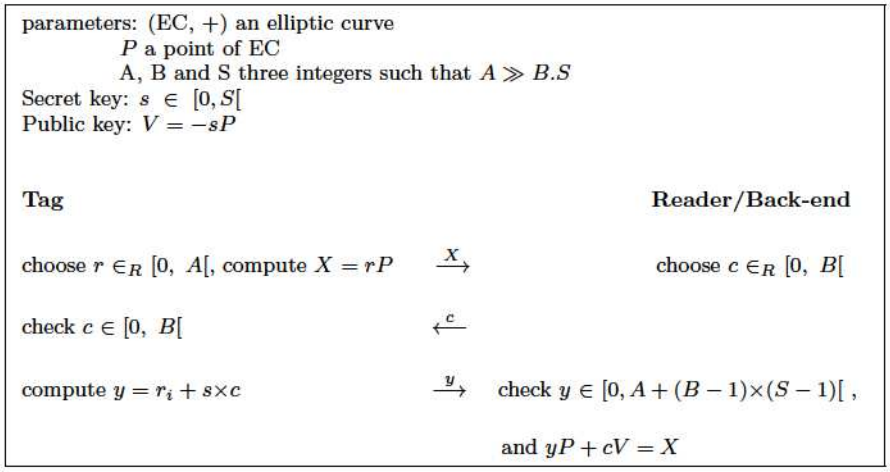


Figure 4.3: Elliptic curve variant of GPS [21]

However, the GPS scheme cannot support the privacy even if the tag does not send back its public key to reader because if the attacker interrogates the tag using the same challenge $c$

he can calculate $yP - X = c(sP)$ which is a constant characterizing each tag. So, a malicious traceability of the tag is made possible.

In [9] Bringer et al. propose a private variant of GPS entitled randomized GPS as demonstrated in Figure 4.4. The difference with the original GPS is that randomized GPS requires two scalar-point multiplications instead of one in the GPS scheme. The computation of the additional elliptic curve point $X'$ from the verifier public key $U = tP$ ensures that only the verifier who has the private key $t$ can make the verification.

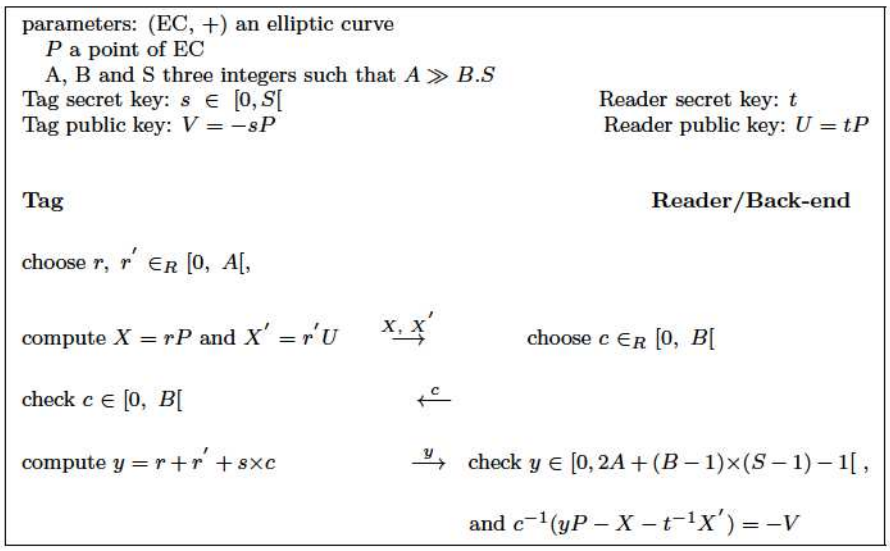


Figure 4.4: Elliptic curve variant of randomized GPS [9]

However, GPS and randomized GPS can not be implemented on low-cost RFID tags due to the significant required resources for scalar-point multiplication.

In [19], Girault proposes a GPS based storage-computation trade-off (next referred to as GPS with coupons for short). As described in Figure 4.5, this approach consists in storing $n$ *coupons* of pairs of numbers $r_i, x_i$ (or in storing only $x_i$ if $r_i$ is regenerated on demand locally) for $0 \leq i \leq n-1$ on the tag with the tag-specific random secret key $s$. In this case, the computation by the tag is limited to $y = r_i + s \times c$. However, this adaptation does not support a moderate security level. An attacker can make a denial of service attack by interrogating the tag more than $t$ times because the number of coupons stored on the tag is limited for cost reasons.
Note that, the storage capacity is the most expensive part of the hardware, so the number of coupons is particularly limited. Moreover, GPS with coupons inherits the problem of privacy of GPS.
The GPS with coupons has been widely studied in the literature and different implementations have been proposed [20], [44]. In [38], McLoone et al. propose an implementation that requires

only 1000 GEs in less than 150 clock cycles using challenges with a low Hamming weight.

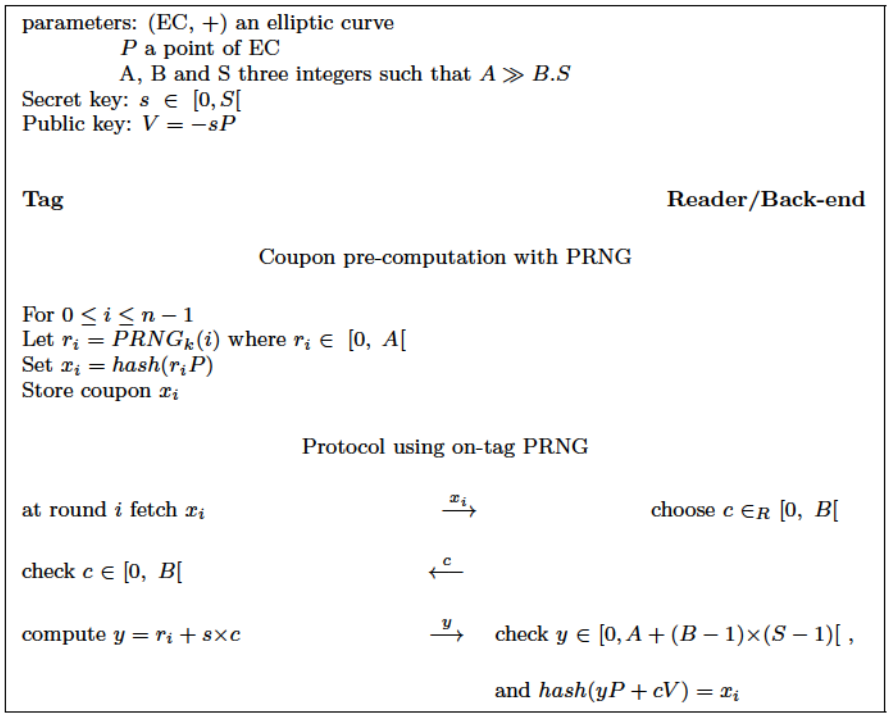


Figure 4.5: Elliptic curve variant of GPS with coupons [19]

## 4.2 The proposed identification approaches

The proposed approaches are storage-security trade-offs varieties of GPS and randomized GPS specially designed for low-cost RFID tags. The main advantage of our approaches over GPS with coupons is that the privacy of the proposed GPS-based approach can be guaranteed and the fact that the storage overhead is supported by the back-end and not the tag. Therefore the number of coupons can be hundred times or more the number of stored coupons of the Girault's approach (GPS with coupons) while reducing the cost of the tags.

In our approaches, tags are only required to implement a deterministic pseudo random number generator (PRNG) with a random secret *seed* (as in GPS with coupons) and simple integer operations (addition and multiplication), to store one random secret key $s$.

Note that for each tag, the coupons stored on the back-end are calculated by a third party -for example- using the same PRNG as on the tag and with the same random secret *seed*.

### 4.2.1 GPS+: the GPS-based approach

The reader implements the elliptic curve scalar-point multiplication and stores for each tag its public key $V = -sP$ and its $n$ *coupons* $r_iP$ (commitments) for $0 \leq i \leq n-1$.

As described in Figure 4.7, upon detecting the tag, the reader generates and sends back to the tag a new challenge $c \in [0,\ B[$.

Upon receiving the reader's challenge, the tag checks that $c \in [0,\ B[$ and re-generates a new pseudo random value $r_i \in [0,\ A[$ where $0 \leq i \leq n-1$; it calculates and sends to the reader the response $y = r_i + s \times c$. As such, the tag only does one pseudo random value generation, one multiplication of two integers, and one addition of two integers.

We recall that the tag uses a deterministic pseudo random value generator with a random secret *seed*.

The reader/back-end searches in its database for a public key $V$ satisfying both $yP + cV = (r_i + s \times c)P - c(sP) = r_iP$ and the matching of the resulting $r_iP$ with a coupon (commitment) associated to the public key of the tag (the result is in the database $DB$). If so the tag is identified. Otherwise the back-end/reader aborts the session concluding that the tag is illegitimate.

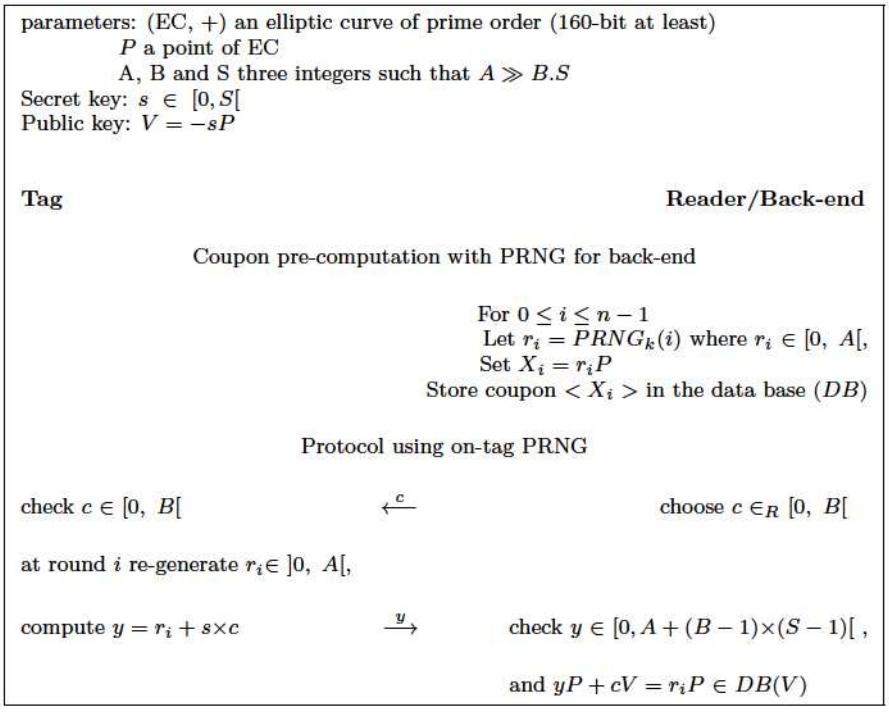


Figure 4.6: GPS+ Proposal: A GPS-based back-end Coupons Identification

The specifications of GPS state that the parameters are typically set to:

- $s$ is at least 160-bit long (i.e. $S \geq 2^{160}$)

- $c$ is 16, 32, 64-bit long ( i.e. $B = 2^{16},\ 2^{32}$ or $2^{64}$)
- $\gg$ means "64 or 80-bit more", $r_i$ is a 240, 256 or 288-bit long (i.e. $A = 2^{240},\ 2^{256}$ or $2^{288}$)

Note that if a low Hamming weight challenges are used to optimize the implementation, the challenge $c$ can be rather longer, about 860 bits as demonstrated in [38].

### 4.2.2 GPS++: the randomized GPS-based approach

As described in Figure 4.7, this approach is relatively similar to the previous one (GPS+) and it is private even if the pre-computed coupons are known to the adversary as we will demonstrate in Section 4.3. However, the server storage is twice the size of the GPS+ and the tag has to generate two pseudo random numbers instead of one.

The protocol starts with the verifier generating a random challenge $c$ which is a random number limited by the security parameter B. The tag generates two pseudo random numbers $r_i$ and $r_i^{'}$ and sends back to reader the response $y = r^{'} + r_i^{'} + s{\times}c$ where $s$ is its secret key limited by the security parameter $S$. Upon receiving, the response, the reader searches for a coupon $< X,\ X^{'} >$ and verifies that $-c^{-1}(yP - X - t^{-1}X^{'})$ is a tag public key associated to the coupon $< X,\ X^{'} >$ in the database. Indeed:

$$\begin{aligned} -c^{-1}(yP - X - t^{-1}X^{'}) &= -c^{-1}((r_i + r_i^{'} + s{\times}c)P - r_iP - t^{-1}(r_i^{'}tP)) \\ &= -c^{-1}((r_i^{'} + s{\times}c)P - r_i^{'}P) \\ &= -sP = V. \end{aligned}$$

Note that, only the verifier that owns the private key $t$ can perform this verification.

## 4.3 Security analysis

In the proposed protocols, we just remove the first message in GPS and randomized GPS schemes, so the security is not weakened and the zero-knowledge property is retained.

The following analysis is valid for both proposed approaches; GPS+ and GPS++. We only consider vulnerabilities over the reader-tag channel. The channels between the reader and the back-end are considered as safe as both equipments have computing and battery resources, they are under the same administrative domain and they can implement any security protocols.

The attacker is assumed behaving according to the Dolev-Yao model, i.e. having full control over the wireless channel (cf. Section 1.2).

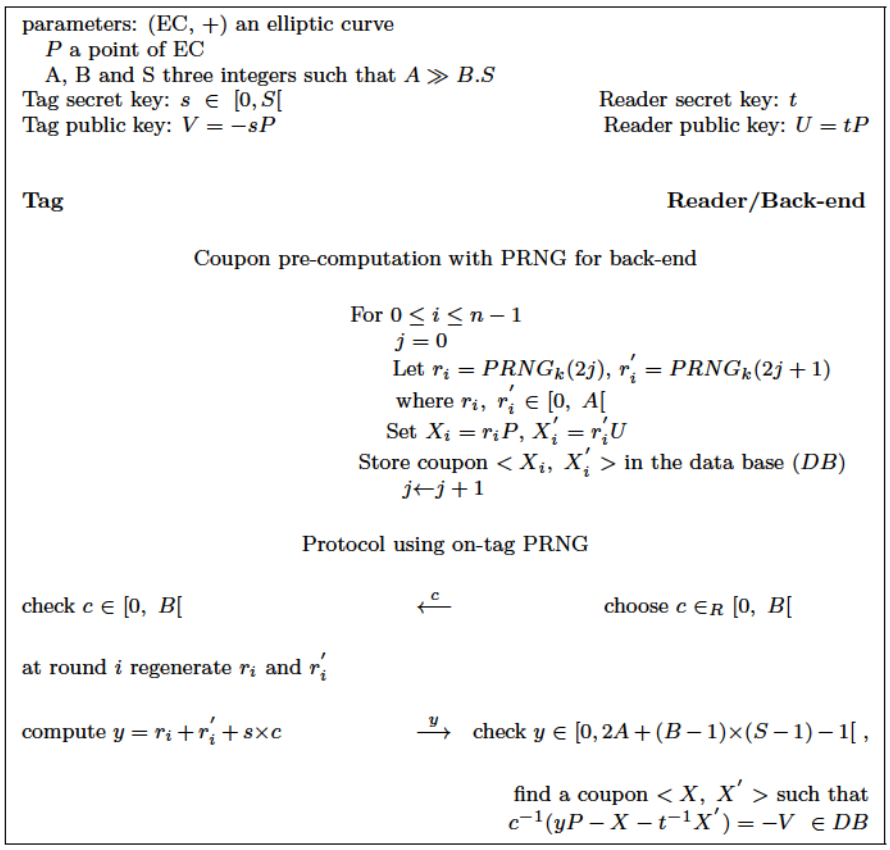


Figure 4.7: GPS++ Proposal: A randomized GPS-based back-end Coupons Identification

### 4.3.1 Resistance to replay attacks

For each identification session, reader and tag generate new pseudo-random values $c$ and $r_i$, respectively; thus making messages randomized (personalized) for each session.

As demonstrated in [21], the probability of impersonation is $1/B^l$ where $l$ is the number of the protocol rounds, and it only depends on the challenge $c$. A replay attack to the reader is unlikely to occur, as the reader/back-end is assumed implementing a good pseudo-random value generator. As such, they are unlikely to generate the same challenge $c$. This probability is $1/B$, and to achieve a higher security level the reader can repeat the protocol $l$ times such that $B^l \geq 2^{80}$. If $B = 2^{32}$, 3 rounds of the protocol achieve a security level higher than 80 bits. However, in many RFID applications, $l$ will often be equal to $l = 1$.

### 4.3.2 Resistance to man-in-the-middle attack/impersonation attacks

For the attacker, the best he can do is experimenting a replay attack on the reader. Indeed, suppose that the attacker wants to be identified as a legitimate tag, he has to produce a valid response $y'$ or to retrieve the tag secret key $s$.

However, the attacker cannot produce a valid response $y'$ from the previously exchanged messages between the tag and readers because any modification in a previous tag's response will be detected in the attacker's response $y'$ thanks to the challenge $c$ generated by the reader itself as showed in [21, 9].

The attacker also can not retrieve the tag secret key $s$ from the exchanged messages between the tag and the reader because the discrete logarithm problem over elliptic curves is supposed to be intractable and $A \gg B.S$ [21]. As such, our approach is resistant to man-in-the middle attacks.

## 4.4 Privacy analysis

The goal of the attacker is to obtain information about a tag $\mathcal{T}$ or trace it. To trace $\mathcal{T}$, the attacker must be able to distinguish between $\mathcal{T}$ and another legitimate tag $\mathcal{T}'$. In the following, we introduce a model that is widely adopted for privacy evaluation.

We demonstrate that the proposed approaches guarantee the tag's privacy according to the most commonly used privacy model proposed by Juels and Weis [31].

This model is defined by an experiment $Exp^{Priv}_{R,A}$ between a challenger $\mathcal{C}$, and adversary $\mathcal{A}$ who can compromise all tags in the system and he has to distinguish between two uncorrupted tags. The adversary $\mathcal{A}$ also cannot make more than a limited number of interactions with the system in each phase (Learning and Challenge, Figure 4.8), and he cannot communicate and compute more than $k$ overall steps where $k$ is the global security parameter of the RFID system $\mathcal{R}$.

In the learning phase of the experiment, the adversary $\mathcal{A}$ is able to corrupt any tag in the system, leaving at least two uncorrupted tags.

In the challenge phase of the experiment, the adversary $\mathcal{A}$ selects two uncorrupted tags as challenge candidates. The challenger $\mathcal{C}$ removes these two tags from the tags set, selects randomly and presents one of them to $\mathcal{A}$ (as a tag oracle). $\mathcal{A}$ interacts with the whole system and corrupts any tag except the challenge tag (tag oracle).

The adversary $\mathcal{A}$ wins if he distinguishes the challenge.

A protocol is private if

$| Pr(Exp^{Priv}_{R,A}\ succeeds\ in\ guessing\ b) - \frac{1}{2} | \leq \epsilon(k)$ where $\epsilon(.)$ is a negligible function.

We suppose that the number of coupons stored on the back-end is at least twice more than the number of interactions possible to the adversary $\mathcal{A}$, this is a realistic assumption as coupons

Experiment $Exp^{Priv}_{R,A}$
**Setup:**
1. The challenger $\mathcal{C}$ initializes the RFID system $\mathcal{R}$ with $n$ tags.
**Phase 1 (Learning):**
2. $\mathcal{A}$ interacts with the whole system and corrupts arbitrary any ($n$-2) tags.
**Phase 2 (Challenge):**
3. $\mathcal{A}$ selects two uncorrupted challenge tags $\mathcal{T}$ and $\mathcal{T}'$ and gives them to $\mathcal{C}$.
4. Let $b \in_R \{0, 1\}$, $\mathcal{C}$ assigns $\mathcal{T}_b = \mathcal{T}$ and $\mathcal{T}_{b\oplus 1} = \mathcal{T}'$ and removes both of these from the current tag set.
5. $\mathcal{A}$ interacts with the whole system including $\mathcal{T}_0$ and $\mathcal{T}_1$ and corrupts arbitrary any tag in the tag set except $\mathcal{T}_b$.
6. $\mathcal{A}$ outputs a guess bit $b'$.
**Exp** succeeds if $b = b'$

Figure 4.8: Privacy experiment

are stored on the back-end and not on the tag, and the size of each coupon can only be 161 bits (the abscissa of the elliptic curve point and one bit for selecting the ordinate).

For GPS+, we suppose that the attacker does not know the tag coupons $r_iP$ for $0 \leq i \leq n-1$, otherwise the tag's privacy cannot be guaranteed as it is the case for GPS (cf. Section 4.1.2), where the tag public key is known to the attacker.

As we have demonstrated in Section 4.1.2, the GPS protocol cannot be private, but GPS+ can be private as we will demonstrate.

In order to demonstrate that GPS+ is private, the adversary $\mathcal{A}$ and the challenger $\mathcal{C}$ execute the privacy experiment as described in Figure 4.8 , except that, $\mathcal{C}$ replaces the tag $\mathcal{T}_b$ in phase 2 of the privacy experiment $Exp^{Priv}_{R,A}$ by a simulator $Sim$ who does not know any secret. We demonstrate that the interaction of $\mathcal{A}$ with $Sim$ will be computationally indistinguishable from an interaction with $\mathcal{T}_b$. In other words, the adversary $\mathcal{A}$ can not detect the presence of $Sim$ and the absence of the challenge tag.

Let $Y_{Sim}$ the set of responses of $Sim$ collected by $\mathcal{A}$ in the phase 2, $Y$ and $Y'$ the set of responses of tags $\mathcal{T}$ and $\mathcal{T}'$, respectively; collected by $\mathcal{A}$ in the phase 1. In order to compromise the privacy (detect the presence of $Sim$), $\mathcal{A}$ has to find some values $y \in Y_{Sim}$ which is invalid for both $\mathcal{T}$ and $\mathcal{T}'$, a necessary condition for this, $\mathcal{A}$ must distinguish between the response of a legitimate tag and a simulator.

The message $y = r_i + s \times c$ (tag's response in GPS+) is statistically indistinguishable if $SB/A$ is negligible [21]. This means that the communication between $\mathcal{A}$ and the tag $\mathcal{T}_b$ can be simulated. In other words $\mathcal{A}$ cannot distinguish between $\mathcal{T}_b$ and $Sim$.

So, the probability that $\mathcal{A}$ detects the presence of $Sim$ i.e. the absence of $\mathcal{T}_b$, is negligible. This means that, the probability that $Pr(Exp^{Priv}_{R,A}$ *succeeds in guessing* $b)$ is close to 1/2. As such, GPS+ supports privacy.

For GPS++, the coupons $< X_i,\ X_i^{'} >,\ i = 1, ..., n$, can be publicly known because randomized GPS is private as demonstrated in [9]. The computation of the additional elliptic curve point $X_i^{'}$ in the commitment (coupon) ensures that only the legitimate reader who has the private key $t$ can perform the verification i.e. find the public key which characterizes each tag. In the response $r_i + r_i^{'}$ hides the tag secret key $s$ in the same way as in GPS.

## 4.5 Performance evaluation

The proposed approaches are only two steps specially designed for low-cost RFID tags. All the complex tasks such as scalar-point multiplication and coupons storage are done at the server (back-end). The tag implements only pseudo random value generator and simple integer operations including: integer multiplication and addition.

As GPS+ approach requires fewer resources on the tag than GPS with coupons (no coupons storage on the tag), it can be implemented in less than 1000 $GEs$ using low Hamming weight challenges, as demonstrated by McLoone et al. [38]. The GPS++ would require a similar amount of resources on the tag as GPS+ because it does not need any additional primitives, only few clock cycles more.

On the other hand, the reader/back-end implements a pseudo random number generator, scalar-point multiplication, and stores one public key and $n$ *coupons* for each tag.

The identification is asymmetrical as the back-end knows the public key $V$ of the tag and not the private key $s$. The back-end stores for each tag the public key and a set of *coupons*, apart from the storage overhead of *coupons*, the cost of identification is almost thus of a conventional symmetric identification (cf. Chapter 1, Figure 1.2), for example in GPS+ approach the back-end has to find first a public key $V$ satisfying both $yP + cV = (r_i + s \times c)P - c(sP) = r_iP$ and the matching of the resulting $r_iP$ value with a coupon associated to the public key $V$. So compared to symmetrical identification, the additional overhead is only to find the coupon $r_iP$ among $n$ *coupons* that are associated to the public key $V$ which is negligible.

To give an overview of our storage-security trade-off approach, we suppose that the time to interrogate a tag is 300 ms and we show in table 4.1 the storage capacity required on the back-end for each tag according to the number of coupons and the time before any temporary desynchronization between the tag and the reader after consumption of all available coupons if the tag is interrogated continuously by an attacker. We note that this time becomes large

enough if the number of coupons is important.

To increase the number of available coupons and to reduce storage overhead on the server, consumed coupons can be deleted and replaced with new coupons (next coupons) computed for example by trusted third party (authority, manufacturer, etc.). Recall that the tag uses a deterministic pseudo random number generator with a random secret *seed* as in GPS with coupons.

| number of coupons | set of tags | storage overhead | | time before desynchronization |
|---|---|---|---|---|
| | | GPS+ | GPS++ | |
| 100 | 100 | 196,5 KB | 393 KB | 30 s |
| 1.000 | 1.000 | 19,19 MB | 38,38 MB | 300 s |
| 10.000 | 10.000 | 1,87 GB | 3,74 GB | 3000 s |
| 100.000 | 100.000 | 187,42 GB | 374,84 GB | 8,33 h |
| 1.000.000 | 1.000.000 | 18,3 TB | 36,6 TB | 83,33 h |

Table 4.1: Overview over proposed approaches

Coupon size can be further reduced under some assumptions on the adversary capabilities by using r-collision resistance hash function i.e. a hash function for which it is difficult to find r distinct inputs which have the same output. So that 200 coupons can be stored in 1 KB as demonstrated in [19]. The idea is that, a zero-knowledge protocol requires in the commitment step, a one-way function such as a hash function. On the other hand, the collision resistance for a hash function, is a stronger property than one-wayness, and it is sufficient to ensure soundness property in zero-knowledge schemes as demonstrated by Girault and Stern [22]. If the length of the challenge sends by the verifier is increased and a r-resistance ($r \geq 3$) hash function is used to compute commitment in a zero knowledge scheme based on discrete logarithm, then le length of the hash function output can be reduced.

## 4.6 Conclusions

In this chapter, we presented two zero-knowledge identification approaches GPS+ and GPS++ based on GPS and randomized GPS; respectively, especially designed for low cost RFID tags. These solutions satisfy the security and privacy requirements for low-cost RFID systems, and they rely on the use of back-end coupons in order to minimize the on-tag dedicated security resources. They benefit from the GPS scheme features like high security level, zero-knowledge, and low complexity. They provide remarkable properties such as privacy and low complexity, and resistance to known classical security attacks. GPS+ has the advantage that any adversary

who has coupons can identify the tag, which making it no private when coupons are known to the adversary. Furthermore, the proposed approaches can be implemented efficiently into low-cost tags in less than 1000 $GEs$.

# Chapter 5

# Conclusion

We investigated, in this work, various challenges in designing lightweight RFID authentication schemes. We presented a ns-3 simulator module for the universal low-cost RFID standard EPC Class-1 Generation-2 (EPCGen2), then we propose three identification protocols based on public key cryptography: a scalable authentication protocol based on NTRU public key cryptosystem, and two zero-knowledge approaches based on GPS and randomized GPS schemes.

In Chapter 2, we introduced the EPCGen2 standard and showed the shortcomings of existing RFID simulators, then we presented our RFID simulator module for ns-3. We showed that, the symmetrical key cryptography is excluded from being used in EPCGen2 standard or in any future scalable standards for low-cost RFID systems when the privacy is needed due to the very short identification time for large tag populations. That does not give any other alternatives than designing a lightweight approaches based on public key cryptography.

In Chapter 3, we investigated the scalability in low-cost RFID systems and we introduced works pertaining to it in the literature. Then we proposed an adaptation of NTRU public key cryptosystem for constrained devices, and a scalable authentication protocol based on this NTRU's adaptation. The proposed authentication protocol inherits from NTRU the high security level, fast encryption and decryption. It can be implemented efficiently into low-cost tags, as tags are only required to implement a lightweight hash function, addition, and bit-wise operations.

In Chapter 4, we considered the use of zero-knowledge schemes in low-cost RFID systems. Zero-knowledge schemes have a reputation of being computationally more expensive than conventional security schemes. We presented two zero-knowledge identification approaches specially designed for low-cost RFID tags. We make use of coupons stored on the back-end in order to

minimize the on-tag required resources. First, we proposed an identification approach based on GPS scheme. The advantages of this approach is that, any verifier who has coupons can identify the tag which does not make this approach private when coupons are known to the adversary. Second, we proposed an identification approach based on randomized GPS, this approach supports the privacy even if coupons are known to the adversary because only the verifier who has a specific private key can identify the tag.

In the future, we shall focus on the following perspectives:

- Investigate the use of other public key cryptography techniques in low-cost RFID which is one of the most important direction for further research. The code based cryptography is one of them and requires defining new codes and techniques that lead to small size of public key. Indeed, the code-based public key cryptography requires only simple bitwise operations. Today, the only drawback that prevents its implementation on RFID is the very large size of the public key.

- Design of algorithms for dynamic adjust of reader configuration in order to accelerate the identification for large tag populations, for example by estimating the number of tags to be identified and thus use the theoretical maximum throughput of framed slotted Aloha.

- Integrate the simulator with existing RFID applications (middleware) which allows to test many aspects related to these applications without need for a hardware (RFID tags and readers). Such approach allows to test performances of the proposed authentication protocols.

- Extend the simulator to support all EPCGen2 mechanisms such as the possibility of writing to the tags as it impacts the identification time due to additional interactions between the reader and the tags.

- Make simulation possible with multiple readers as provided by the standard in order to accelerate the identification, and simulate more complex and realistic scenarios.

# Appendix A

# Methods description

| Class | Method | Description |
|---|---|---|
| RfidChannel | Send | Broadcast the packet to all equipments in the antenna range. |
| | Add | Add all physical layers to be known by the channel in order to have the reference of all possible destinations. |
| RfidPhy | Send | Send the packet in order to be transmitted by the channel to other equipments. |
| | StartRecv | The physical layer receives a packet. |
| | EndRecv | This method is called when the receiving procedure is done without a collision. |
| RfidPhyState | GetState | Return the physical layer state: idle, receiving or sending packet. |
| | SwitchToTx | Switch the physical layer state to "sending". |
| | SwitchToRx | Switch the physical layer state to "receiving". |
| RfidIdentification | Send | Return if the packet is sent without error or not. |
| | Receive | This method is called when a packet is received by the physical layer. |
| | SetEquipementState | Set the new state of the equipment after receiving a packet and the decision that should be taken after. |

| | | |
|---|---|---|
| RfidReaderIdentification | SetQForQuery | Generate a random number to be sent in query message. |
| | SetQForQueryAdjust | Modify the initial random value. |
| | SendQueryRepOrAdjust | If timeout is exceeded, this method will be automatically called to send QeryRep or QueryAdjust and that's according to the equipment state. |
| RfidTagIdentification | SetInitialConfiguration | Set the initial configuration when the tag is powered up. |
| | SetResponseToQuery | Set the adequate response after receiving a query message. |
| | SetResponseToAck | Set the adequate response after receiving an ACK. |
| RfidConfiguration | SetData0 | Set the Data0 duration. |
| | GetData1 | Return the Data1 duration. |
| | GetPreambleDuration | Return preamble duration. |
| | GetTagPreamble | Return tag preamble duration. |
| EpcMemory | GetEpc | Return 96 bits EPC identifier . |
| | GetStoredCrc | Return stored CRC16 calculated from EPC and stored PC. |
| | GetStoredPc | Return stored PC value. |
| EpcHeader | Serialize | Store a header into the byte buffer of the packet. |
| | Deserialize | Re-create a header from the byte buffer of the packet. |
| | GetSerializedSize | Return the number of bytes which are needed to store the full header data by Serialize. |
| RfidNetDevice | SetChannel | Set the channel used to transmit information. |
| | SetPhy | Set the physical layer. |
| | SetIdentification | Set the identification layer. |

# Appendix B

# Publications

## Conferences

- Ethmane El Moustaine, Maryline Laurent, "A Lattice Based Authentication for Low-Cost RFID", Third IEEE RFID Technology and Applications (IEEE RFID-TA 2012), Nice, France, 5-7 November, 2012.
- Ethmane El Moustaine, Maryline Laurent, "GPS+: a back-end coupons identification for low-cost RFID", 6th ACM Conference on Security and Privacy in Wireless and Mobile Network (WISEC 2013), Budapest, Hungry, 17-19 April, 2013.

## Journals

- Laurent Gomez, Maryline Laurent, Ethmane El Moustaine, "Risk Assessment along Supply Chain: A RFID and Wireless Sensor Network Integration Approach", Sensors Transducers journal (ISSN 1726-5479), Vol.14-2, Special Issue, March 2012, pp.269-282.
- Ethmane El Moustaine, Maryline Laurent, "An RFID module for the ns-3 network simulator", Computer Standards & Interfaces journal, submitted.
- E. El Moustaine, M. Laurent, "Les systèmes RFID : la technologie, les risques et les solutions de sécurité", Techniques de l'Ingénieur, Sécurité des systèmes d'information, H5325, 2012.

## Patents

- E. El Moustaine, M. Laurent, "Procédé pour crypter des données dans un cryptosystème NTRU (N,p,q)", numéro le n° PCT/IB2013/055122 (21 juin 2013), dépôt en France sous 12 55948 le 22 juin 2012.

- E. El Moustaine, M. Laurent, " RFID bas-coût", numéro d'enregistrement 11 51399, février 2011.

## Workshops

- L. Gomez, M. Khalfaoui, E. El-Khoury, C. Ulmer, J.-P. Deutsch, O. Chettouh, O. Gaci, H. Mathieu, E. El Moustaine, M. Laurent, H. Schneider, C. Daras, A. Schaad, RESCUEIT : sécuRisation dE la Chaîne logiStique orientée serviCe depUis le mondE des objets jusqu'à l'univers InformaTique , Workshop Interdisciplinaire sur la Sécurité Globale (WISG'11), Troyes, France, Jan. 25-26 2011.

- J.-P. Deutsch, J. Hue, Y. Gaffé, L. Gomez, M. Khalfaoui, M. Laurent, A. Levieux, E. El Moustaine, RESCUEIT: sécuRisation dE la Chaîne logiStique orientée serviCe depUis le mondE des objets jusqu'à l'univers InformaTique , Workshop Interdisciplinaire sur la Sécurité Globale (WISG'13), Troyes, 22-23 janvier 2013.

- L. Gomez, M. Laurent, E. El Moustaine, " Integration of RFID and Wireless Sensor Networks into a Supply Chain Management System", The First International Workshop on Sensor Networks for Supply Chain Management, WSNSCM 2011, Nice, France, ISBN 978-1-61208-144-1, August 21-27 2011.

# Appendix C

# Résumé

La technologie d'identification par radio fréquence, souvent désignée par l'abréviation RFID, est l'une des technologies les plus prometteuses dans le domaine de l'informatique ubiquitaire. En effet, elle pourrait bien transformer les processus d'identification parce qu'elle offre de nombreux avantages par rapport à d'autres systèmes d'identification grâce à ses caractéristiques uniques qui permettent de scanner et d'identifier une étiquette passive sans avoir besoin de contact visuel (sans ligne de vue directe) ou physique.

Aujourd'hui, la technologie RFID est très utilisée dans les processus de fabrication, la logistique, les transports publics, le contrôle d'accès physique, les appareils médicaux embarqués, l'identification des animaux, etc.

Les systèmes RFID sont constitués de trois éléments principaux: étiquette, lecteur et éventuellement une base de données comme le montre la Figure C.1.


Figure C.1: Un système RFID [23]

La plupart des étiquettes RFID actuellement déployées n'ont pas de batterie et sont alimentées par un champ d'induction par résonance magnétique généré par le signal du lecteur. Ceux-ci sont connus comme des étiquettes RFID passives. Les étiquettes actives sont celles qui portent une batterie et fonctionnent indépendamment du signal du lecteur. Ce type d'étiquettes

est coûteux, elles sont réservées aux applications haut de gamme. Les étiquettes hybrides, appelées semi-actives, utilisent leur batteries exclusivement pour effectuer des opérations internes, mais elles utilisent le signal du lecteur pour la communication. Ce type de RFID est une combinaison de technologies RFID actives et passives.
La terminologie "étiquettes RFID à bas coût" est largement utilisée pour désigner des étiquettes RFID passives à très faibles coûts.
Aujourd'hui, le faible coût de la technologie RFID, les énormes enjeux économiques qu'elle présente et la facilité d'utilisation, font de cette technologie de la prochaine révolution technologique après l'Internet.

## C.1 Les défis d'authentification dans les systèmes RFID bas coûts

Dans le domaine de la sécurité de l'information, l'authentification nécessite l'utilisation des algorithmes cryptographiques qui nécessitent des ressources importantes en termes de calcul, énergie et mémoire.
Comme l'identification est le fondement même de la technologie RFID, cette identification doit être sécurisée pour éviter l'usurpation d'identité d'étiquette et/ou du lecteur. Cependant, pour rendre possible le déploiement des RFID à grande échelle, deux conditions doivent être remplies. Tout d'abord, un très bas coût des étiquettes doit être garanti, ce qui conduit inévitablement à des dispositifs de très faibles capacités qui ne peuvent pas effectuer des calculs complexes. Deuxièmement, les problèmes liés à la scalabilité (passage à l'échelle) doivent être résolus lorsqu'une lecture massive des étiquettes est nécessaire.
Il n'est donc pas surprenant que l'inconvénient de la technologie RFID pourrait être liée à la sécurité et les risques sur la vie privée, rendant ainsi difficile la résistance aux attaques standards qui sont: les attaques par rejeu, l'homme du milieu, déni de service, etc. L'effet que les étiquettes répondent à toutes les requêtes du lecteur, les problèmes liés à la vie privée sont difficiles et sont liés au suivi clandestine de l'étiquette et la confidentialité persistante (forward secrecy). De façon très simple, un protocole d'identification RFID est vulnérable au suivi clandestine quand un adversaire est capable de reconnaître une étiquette RFID vue précédemment. La confidentialité persistante est la propriété qui garantit que la divulgation de l'information secrète d'un tag ne révèle pas les secrets utilisés dans les sessions précédentes.

Aujourd'hui, ces problèmes constituent un défi majeur pour les chercheurs, exigeant de pousser les limites de la cryptographie. Ces travaux de recherche sont encouragés par la Commission européenne. En 2008, un projet de recommandation [15] a été publié sur la sécurité et les problèmes liés à la vie privé dans les systèmes RFID indiquant que les applications RFID doivent fonctionner de manière sûre et que la recherche doit aboutir à des solutions de sécurité performantes de faible coût pour les dispositifs RFID.

La sécurité et les problèmes de la vie privée dans les systèmes RFID concernent principalement les communications sans fil entre l'étiquette et le lecteur, tandis que les communications entre le serveur (base de données) et le lecteur sont supposées assurées parce que ces dispositifs ont les capacités de calcule et l'énergie nécessaires pour sécuriser leur communications avec des algorithmes de chiffrement symétriques ou asymétriques.

Un autre aspect lié à la technologie RFID tout aussi problématique est la caractéristique "système ouvert" qui caractérise beaucoup des systèmes RFID lorsque le système qui interroge l'étiquette ne peut pas être identifié à l'avance en raison par exemple de la mobilité des étiquettes RFID dans certains cas d'utilisation, autrement dit, il n'y a aucune garantie que le lecteur (vérificateur) ne soit pas malveillant. Dans un tel contexte, les solutions sont des protocoles d'identification à divulgation nulle de connaissance parce que ce type de protocoles ne nécessite pas un partage de secrets entre l'étiquette et le lecteur. Cependant, ces protocoles tels qu'ils sont, exigent une quantité de ressources au-delà des capacités d'étiquettes RFID à bas coûts.

Le schéma communément utilisé pour réaliser l'authentification est appelé challenge/réponse: comme le montre la Figure C.2, le lecteur envoie un challenge C à l'étiquette qui prouve son identité en répondant à ce challenge. Évidemment, l'adversaire ne devrait pas être en mesure de se faire passer pour une étiquette légitime, même si il a espionné toutes les réponses précédentes de l'étiquette. La réponse de l'étiquette au lecteur, consiste à chiffrer le challenge c reçu en utilisant un algorithme de chiffrement E, une clé cryptographique secrète K et éventuellement d'une valeur aléatoire r qu'elle a fraîchement générée.

| **Lecteur** | | **Étiquette** |
|---|---|---|
| choisir $c \in_R \{0,\ 1\}^n$ | $\xrightarrow{c}$ | choisir $r \in_R \{0,\ 1\}^n$, calculer |
| Retrouver $k$, vérifier $E_k(c,\ r)$ | $\xleftarrow{r,\ E_k(c,\ r)}$ | $E_k(c,\ r)$ |

Figure C.2: Un schéma d'identification Challenge/réponse

Ce type de protocole a deux inconvénients majeurs si l'algorithme de chiffrement E est symétrique. L'un d'eux est l'absence de la scalabilité si la totalité de la réponse de la balise est différente à chaque tentative d'authentification, dans ce cas, le système doit effectuer une recherche exhaustive pour trouver la clé secrète K afin d'authentifier l'étiquette. Le deuxième inconvénient est le problème lié à la vie privée c-à-d si un adversaire est capable de reconnaître une étiquette vue précédemment ou si la réponse de l'étiquette est partiellement constante même entre deux sessions d'authentification successives légitimes. Par exemple, l'étiquette utilise un pseudo index qui facilite pour le système la recherche de la clé de l'étiquette et qui sera mit à jours après chaque authentification auprès d'un lecteur légitime, dans ce cas un adversaire peut tracer l'étiquette entre deux sessions d'authentification en utilisant le pseudo index. Il s'agit de l'un des principaux problèmes que la RFID et plus généralement l'informatique ubiquitaire doit résoudre comme Mark Weiser l'avait déjà prédit en 1991 [54].
Une solution qui supporte à la fois la scalabilité et le respect de la vie privée pour des systèmes RFID à bas coût ne peut être basée que sur la cryptographie légère à clés publiques. En effet, si le système RFID implémente un système de chiffrement à clé publique, le lecteur n'a qu'une seule clé privée pour déchiffrer la réponse de toute étiquette même si cette réponse est différente à chaque tentative d'authentification.
Cela résout les problèmes de la scalabilité et garantit la protection de la vie privée. De plus, un haut niveau de protection de la vie privée nécessite l'utilisation de certaines techniques de la cryptographie à clés publiques [52].

Cependant, la plupart des chercheurs croient que la RFID à faible coût ne peut pas profiter de la cryptographie à clés publiques et doit donc être basée sur la cryptographie à clés secrète. Il est donc impossible sous une telle hypothèse qu'un protocole d'authentification puisse garantir la scalabilité et la protection de la vie privé avec un coût raisonnable.

## C.2 Étude de la scalabilité dans les systèmes RFID à bas coûts

### C.2.1 Le standard EPC Class-1 Generation-2

Le standard EPCglobal Class-1 Generation 2, souvent appelé EPCGen2 est un nouveau standard pour la technologie RFID passive à bas coût. Il est considéré comme le successeur du code-barre et il est largement considéré comme le standard universel pour les étiquettes RFID à bas coûts. Il est conçu pour satisfaire les besoins de la chaîne d'approvisionnement.

La petite taille des étiquettes EPCGen2 leur permet à être implantées dans des objets, et l'identification par fréquence permet aux étiquettes non seulement à être lu en grand nombre, mais aussi dans des conditions visuelles et environnementaux difficiles.

Cependant, EPCGen2 ne définit pas exactement comment le lecteur devrait réagir dans des situations critiques comme les collisions ou comment gérer les différentes commandes lors de l'identification des étiquettes. C'est pourquoi un simulateur peut être un outil très utile pour découvrir les comportements de protocole dans différents scénarios.
Nous créons un nouveau module dans ns-3 (Network Simulator 3) pour prédire les performances d'identification pour EPCGen2 protocole de communication dans les différentes configurations. Ce module permet de tester la scalabilité (passage à l'échelle) de cette norme afin d'établir un cadre strict pour une identification sécurisée, par exemple, quelle type d'approche de sécurité pourraient être envisagées malgré que l'EPCGen2 ne définit aucun mécanisme de sécurité. D'autres part, la simulation RFID est très utile pour de nombreuses défis de recherches tels que la sécurité, la protection de la vie privée et l'optimisation du débit sur le canal lecteur-étiquette qui sont difficiles à tester dans la pratique en raison de la nature des étiquettes RFID.

### C.2.2 Généralités sur l'EPCGen2

La norme EPCGen2 définit les caractéristiques physiques et logiques pour les systèmes RFID qui utilisent des étiquettes passives afin de remplacer le traditionnel code-barre. Tableau C.1 résume les principales caractéristiques d'un lecteur et d'une étiquette compatibles avec EPCGen2.

| | |
|---|---|
| Fréquence | Ultra Haut Fréquence: 860 MHz - 960 MHz |
| Porté | Approximativement 5 meters |
| Débit | Liaison aval: 26.7 - 128 kbps |
| | Liaison de rétrodiffusion: 5 - 640 kbps |
| Codage | Liaison aval: PIE (pulse interval encoding) |
| | Liaison de rétrodiffusion: FM0, Miller-modulated subcarrier |
| Modulation | Liaison aval: DSB-ASK, SSB-ASK ou PR-ASK |
| | Liaison de rétrodiffusion: ASK ou PSK |
| Protocole d'accès multiple | Variante de slotted Aloha (Q Protocol) |
| Étiquettes | Passives, EPC, mémoire réservée |

Table C.1: Caractéristiques du standard EPCGen2

L'accès au canal sans fil partagé est basé sur une variante d'Aloha appelée Q protocol. Le

paramètre Q est spécifié dans la commande Query, et il varie entre 0 et 15. Pour répondre à la requête du lecteur, chaque étiquette qui a reçu la commande précédente, choisit un nombre aléatoire dans l'intervalle 0 à $2^Q - 1$ inclus. L'étiquette qui tire une valeur égale à zéro doit répondre immédiatement par une séquence pseudo-aléatoire de 16 bits (RN16). Sinon l'étiquette doit attendre une autre commande de la part du lecteur comme suit:

- Aucun étiquette ne répond: le lecteur peut envoyer une autre commande, une QueryRep ou une QueryAdjust.
- Une étiquette répond: le lecteur peut identifier l'étiquette sans encombre comme le démontré la Figure C.3.
- Plusieurs étiquettes répondent: le lecteur reçoit plusieurs messages RN16 dont le nombre est égal au nombre des étiquettes qui ont tiré une valeur égale à zéro. Le lecteur essaye de résoudre la collision et envoie un acquittement. Si la collision ne peut être résolu, le lecteur envoie une commande QueryRep ou QueryAdjust.

Si aucune étiquette ne répond ou si plusieurs étiquettes répondent en même temps, le lecteur modifie la valeur du paramètre Q envoyée dans la commande Query, en utilisant QueryAdjust. Il diminue la valeur du Q s'il n'a pas reçu de réponse pour augmenter la probabilité d'obtenir une réponse, et en cas de collision, il augmente la valeur du Q pour diminuer la probabilité que deux étiquettes génèrent en même temps une valeur égale à zéro.
Dans le cas où une seule étiquette répond au lecteur, le protocole d'identification est décrit dans la Figure C.3.

Les échanges entre l'étiquette et le lecteur doivent suivre des contraintes de chronométrage très précises comme le montre la Figure C.4.

### C.2.3 Modèle de simulation

Nous avons modélisé la mobilité et l'activation des étiquettes, la propagation du signal entre les étiquettes et le lecteur, la réception et la transmission des signaux au niveau des étiquettes et du lecteur, les commandes spécifiques dans le protocole de communication du EPCGen2, etc. En ce qui concerne les paramètres de chronométrage et de modulation, nous avons adopté un modèle semblable à celui de [18] comme décrit au Tableau C.2.

Selon le modèle le plus utilisé dans l'espace libre [36], le signal reçu est atténué par un facteur

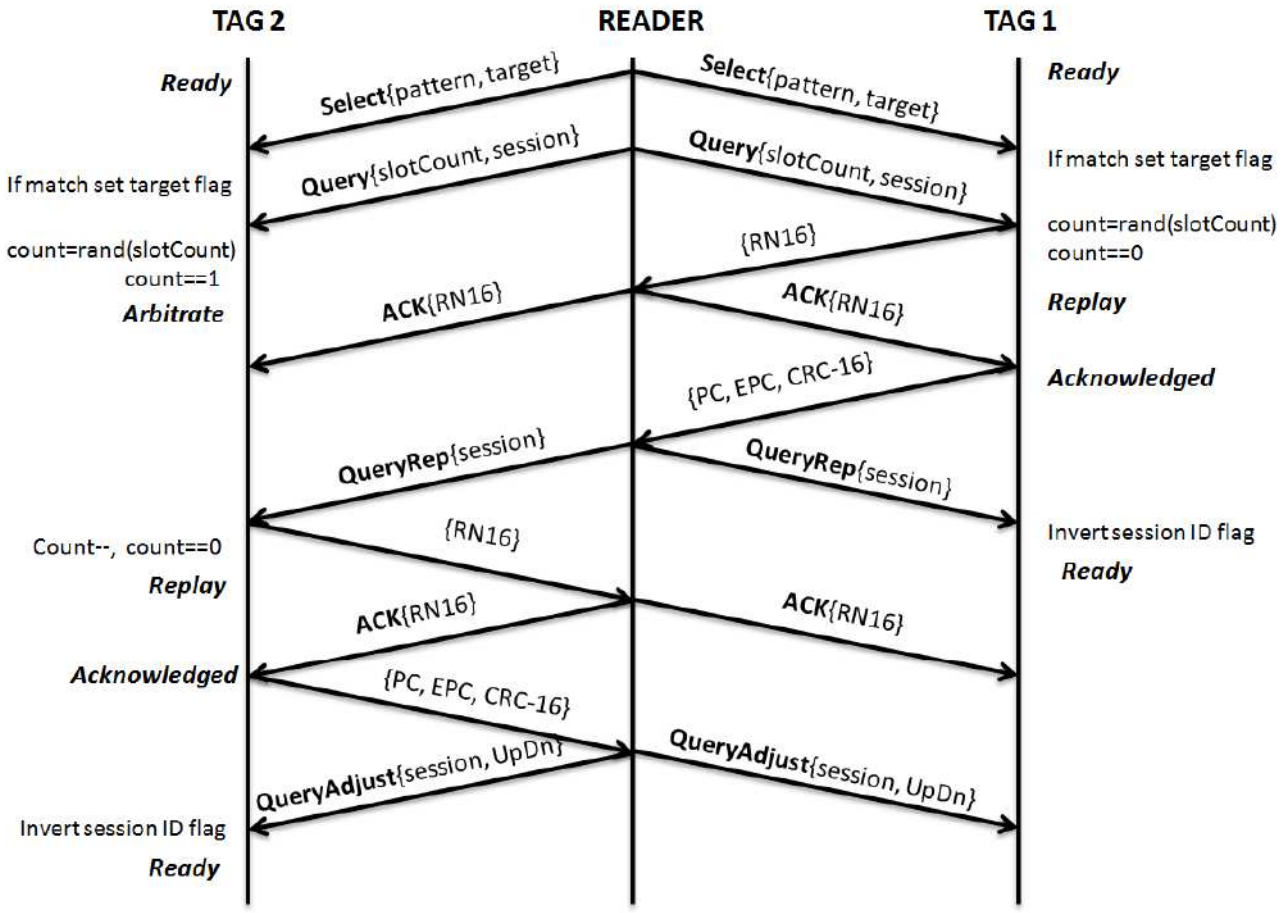


Figure C.3: échange lecteur-étiquettes supportant d'identification

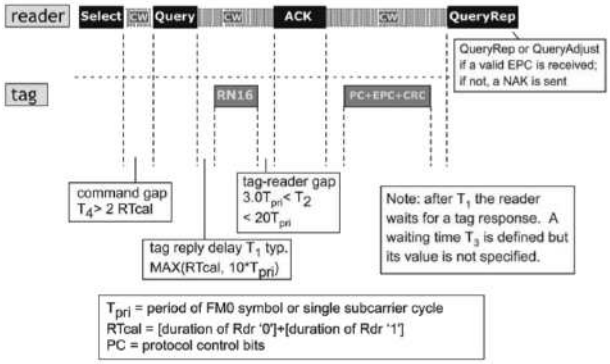


Figure C.4: Contraintes de chronométrage dans EPCGen2 [11]

D qui présente l'affaiblissement de propagation:

$$D = \left(\frac{4\pi r}{\lambda}\right)^2$$

Où r est la distance entre le récepteur et l'émetteur et $\lambda$ est la longueur d'onde. De nombreuses autres paramètres de configurations sur la puissance, la sensibilité etc. sont présentés au Tableau

| Paramètre | Valeur |
|---|---|
| Tari | 8.33 $\mu$s |
| Data-1 | 16.66 $\mu$s |
| TRext | 0 |
| Divide Ratio | 8 |
| Modulation | 2 |
| T1 | 70.7 $\mu$s |
| T2 | 18.7 $\mu$s |
| T3 | 62.5 $\mu$s |
| T4 | 50.0 $\mu$s |

Table C.2: Paramètres de chronométrage

C.3.

| Paramètre | Valeur |
|---|---|
| **Lecteur** | |
| Fréquence | 866 MHz |
| Puissance | 2000 mW |
| Faisceau de l'antenne | 60° |
| Sensibilité | -80 dBm |
| **Étiquette** | |
| Sensibilité | -14 dBm |
| Facteur de rétrodiffusion | 0,25 |
| Vitesse | 1 m/s |
| **Canal de propagation** | |
| Exposant d'affaiblissement de propagation | 2.0 |

Table C.3: Paramètre de configuration

Nous avons défini un algorithm de gestion des commandes comme le montre la Figure C.5, il indique comment le lecteur utilise les différentes commandes: Query, QueryRep et QueryAdjust.

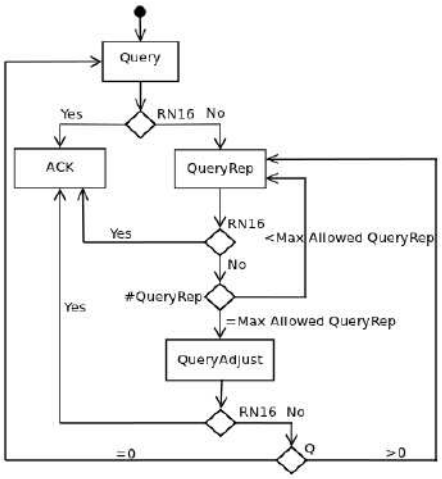


Figure C.5: Algorithme de gestion des commandes

On définit dans la Figure C.6 un algorithme de gestion des collisions.

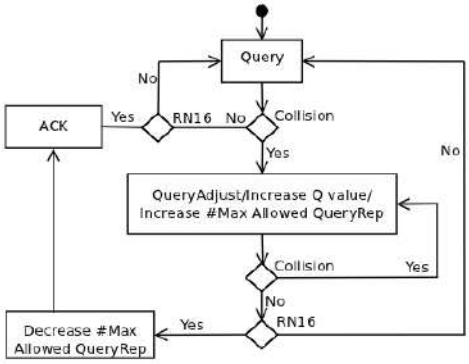


Figure C.6: Algorithme de gestion des collisions

### C.2.4 Conception logicielle

Nous modélisons la couche physique et la couche logique du protocole de communication du EPCGen2. La figure C.7 présente une version simplifiée du diagramme des classes.

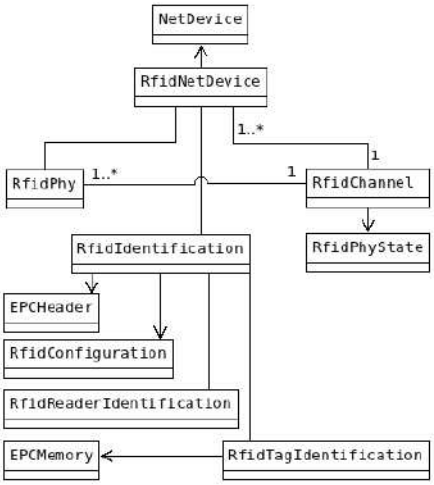


Figure C.7: Diagramme des classes du simulateur

### C.2.5 Les scénarios simulés

Nous simulons et étudions les performances du standard EPCGen2 dans trois scénarios afin d'obtenir la durée d'identification, le nombre de collisions détectées, et l'atténuation de puissance. Dans la pratique, ces scénarios correspondent aux principales étapes de la chaîne d'approvisionnement pour laquelle le EPCGen2 a été conçu: la prise d'inventaire, le contrôle de réception et d'expédition. Ces scénarios sont les suivants:

- Étiquettes fixes: Ce scénario correspond à la prise d'inventaire. Il est caractérisé par un lecteur RFID qui envoie en permanence les commandes Query, QueryRep and QueryAdjust

selon le modèle de simulation (cf. C.2.3).

Les étiquettes sont distribuées aléatoirement dans une palette 1mx1mx1m, située à une distante de 2 mètres du lecteur.

- Étiquettes mobiles: Ce scénario correspond au contrôle de réception et d'expédition. Sa seule différence de avec le scénario précédent est que les étiquettes sont mobiles et elles sont à une distance de 1 mètre du lecteur et elles se trouvent hors de la couverture de l'antenne du lecteur qui le traversent durant la simulation.

- Étude de puissance: Ce scénario examine la puissance reçue par les étiquettes selon la distance qui les sépare du lecteur. Une étiquette est créée à une distance du lecteur et la puissance est mesurée, puis une autre est créée plus loin et la puissance est mesurée à nouveau, etc. ce qui permet d'établir une graphe de puissance en fonction de la distance du lecteur.

### C.2.6 Les résultats de simulation

Afin d'obtenir des résultats significatifs, chacun des scénarios précédents est lancé mille fois ce qui mène à des valeurs moyennes. Nous obtenons le temps d'identification et le nombre des collisions lors de l'identification, comme l'indiqué le Tableau C.4. Pour 1000 étiquettes mobiles, le simulateur a réussi d'identifier uniquement environ 780, et ça c'est à cause de la mobilité et la vitesse des étiquettes.

| Scénario | Nombre des étiquettes | 1 | 10 | 100 | 256 | 500 | 1000 |
|---|---|---|---|---|---|---|---|
| Étiquettes fixes | Identification time (sec) | 0.009 | 0.038 | 0.350 | 0.875 | 1.690 | 2.924 |
| | Number of collisions | 0 | 2.425 | 19.053 | 48.874 | 95.65 | 191.032 |
| Étiquettes Mobiles | Identification time (sec) | 0.40 | 1.293 | 1.482 | 1.629 | 2.174 | 2.980 |
| | Number of collisions | 0 | 0,154 | 12,7 | 53,697 | 97,825 | 152,206 |

Table C.4: Résultats de simulation

La durée d'identification et le nombre de collisions du Tableau C.4 sont présentés graphiquement dans les Figures C.8 et C.9; respectivement.

On remarque que l'identification des étiquettes fixes est achevée dans un laps de temps très court comme le montre la Figure C.8. Ce temps est presque linéaire, la pente diminue légèrement à partir du point de 500 cela est du au fait que la configuration que nous avons choisie est mieux adaptée pour les grands nombres des étiquettes.

L'identification prend plus de temps dans le scénario des étiquettes mobiles, cela est du à la mobilité des étiquettes. En effet, premièrement les étiquettes se trouvent au début du

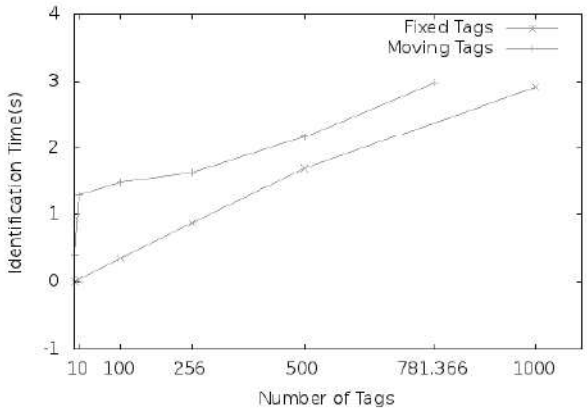


Figure C.8: Durée d'identification

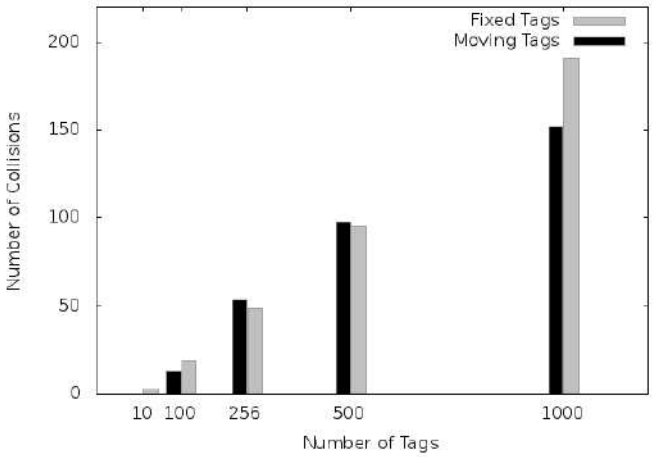


Figure C.9: Collision detection

lancement du scénario hors de la portée du lecteur avant qu'elles la traversent ce qui prend un temps supplémentaire par rapport aux étiquettes fixes. Deuxièmement, lors de l'identification d'un grand nombre des étiquettes mobiles, l'identification se fait groupe par groupe. En effet, un groupe d'étiquettes dans la même région reçoit des commandes Select et Query, dans le même temps, un autre groupe ne reçoit pas ces commandes car il est hors de portée du lecteur; quand il rentre dans le champs de couverture, ce groupe ne répond pas aux commandes QueryRep et QueryAdjust envoyées par le lecteur et reste en attente des commandes Select et Query qui lancent un nouveau cycle, quand ce dernier commence, une partie du groupe précédent sera identifiée de nouveau car elle se trouve toujours dans le champs du couverture du lecteur, ce qui produit plus de collisions pour une grand nombre des étiquettes, et augmente par conséquent le temps d'identification. Cependant, on remarque que pour 1000 étiquettes, les collisions sont plus importantes pour les étiquettes fixes, cela due au fait que le lecteur n'arrive pas à identifier plus que 780 étiquettes à cause de la mobilité.

La Figure C.10 montre que le niveau de puissance diminue rapidement dans les trois premiers mètres qui séparent le lecteur et les étiquettes. Ensuite, la puissance diminue plus lentement. La sensibilité de l'étiquette est définie à -14 dBm et le modèle implémenté par la couche canal permet la communication entre l'étiquette et le lecteur jusqu'à une distance de 6 mètres.

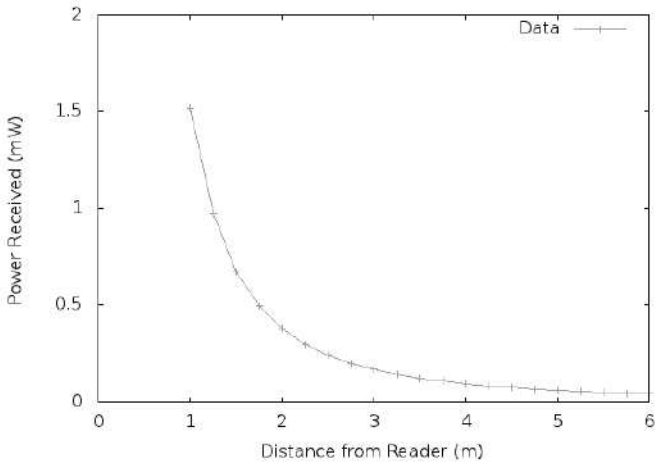


Figure C.10: Atténuation de puissance

Le résultat principal est la grande scalabilité de cette norme c-à-d le temps d'identification très court pour les grands nombres des étiquettes (2.924 s pour 1000 étiquettes mobiles), excluant ainsi toute utilisation du chiffrement symétrique dans tout standard pour les systèmes RFID à bas coût où la scalabilité et la protection de la vie privée sont nécessaires.
Ce module est plus réaliste que les simulateurs existants dans la littérature car il implémente les fonctionnalités les plus couramment utilisées de EPCGen2 avec une adaptation plus facile. Ce simulateur est très utile pour un réglage adéquat du EPCGen2 avant tout déploiement, il est très utile dans de nombreuses défis de recherches liés à la technologie RFID comme la sécurité, la protection de la vie privée, et l'accélération de l'identification des étiquettes. Il sera bientôt disponible en téléchargement libre.

## C.3 L'authentification des RFID à bas coût en utilisant les réseaux

Nous avons démontré par simulation dans C.2 que la cryptographie à clé publique est le mieux adaptée pour beaucoup des applications RFID à bas coûts. Malheureusement, ce type de cryptographie exige une quantité de ressource très importante au-delà de la capacité des étiquettes RFID à bas coûts. Cette thématique n'a pas été bien abordée par les chercheurs. Seulement

quelques solutions ont été proposées dans la littérature [43], [8], [33], [35], [50], [42] et [49], mais la quantité de ressources que ces solutions exigent reste élevée pour qu'elles soient implémentées sur des étiquettes à bas coûts.

Nous proposons une adaptation du célèbre cryptosystème à clés publique NTRU [26] aux équipements embarqués à très faible capacité de calcul, et un protocole d'authentification basé sur cette adaptation spécialement conçu pour des étiquettes RFID à bas coût. Cette adaptation ramène le chiffrement d'une opération mathématique difficile (convolution modulaire) à des opérations simples. En effet, grâce aux propriétés de l'anneau des polynômes dans lequel NTRU opère, nous avons rendu possible le chiffrement par l'étiquette d'un faible volume de données en utilisant uniquement des additions et des décalages circulaires. Le protocole d'authentification proposé garantit la sécurité, la protection de la vie privée, et la scalabilité (passage à l'échelle).

### C.3.1 NTRU

NTRU [26] est un cryptosystème probabiliste à clés publiques très léger, il est considéré sûr par les standards IEEE 1363.1 [3] et X9 [4]. La sécurité de NTRU est basée sur la difficulté de trouver le plus court vecteur et le vecteur le plus proche dans un réseau aléatoire.
Les opérations de NTRU sont réalisées dans l'anneau de polynôme $\Re = \mathbb{Z}[X]/(X^N - 1)$, mais il s'intéresse plus particulièrement aux petits polynômes avec des coefficients dans $\{-1,\ 0,\ 1\}$. NTRU définit quatre sous ensembles de petits polynômes $D_f$, $D_g$, $D_f$, $D_r$ et $D_m$, et deux entiers $p$ et $q$ premiers entre eux, $p$ est généralement égal à 2, mais dans certaines configurations $p$ peut être un petit polynôme.
Pour générer une paire de clés, on génère deux polynômes $f$ et $g$ dans $D_f$ et $D_g$; respectivement, tel que $f$ est inversible modulo $q$ et modulo $p$. Soient $f_q$ l'inverse de $f$ modulo $q$ et $f_p$ l'inverse de $f$ modulo $p$. Alors la clé publique est $h = p{*}g{*}f_q \ modulo\ q$ et la clés privée est $(f,\ g)$

Pour chiffrer un message $m$, on génère aléatoirement un polynôme $r \in D_r$ et on calcule le message chiffré $e = r{*}h + m \ modulo\ q$.

Pour déchiffrer un message $e$, on utilise la clés privée $(f,\ g)$ et on calcule le polynôme $a = f{*}e\ (mod\ q) = p{*}r{*}g + m{*}f\ mod\ q$ et on obtient sa valeur dans l'anneau $\Re$, puis on calcule sa valeur modulo $p$ pour obtenir $a \ modulo\ p = m{*}f \ modulo\ p$ et puis le message $m$ parce que $f$ est inversible modulo $q$.

### C.3.2 L'adaptation proposée

Chaque élément $f$ de l'anneau $\Re$ s'écrit: $f = (f_0, f_1, ..., f_{N-1}) = \sum_{i=0}^{N-1} f_i x^i$

La multiplication dans cet anneau est une convolution modulo $X^N - 1$ et l'addition se fait terme à terme. Cependant, cet anneau est cyclique cela signifie que la multiplication par $X$ est équivalente au décalage des coefficients à droite par une position, le coefficient de $X^i$ devient le coefficient de $X^{i+1}$.

Ce produit de convolution est comme suit:

$$(f*g)(X) = \sum_{i=0}^{N-1} h_i X^i, \forall\ 0 \leq k \leq N-1,\ h_k = \sum_{i+j \equiv k\ mod\ N}^{k} f_i g_j$$

Si $f^{'}$ est un décalage circulaire vers la droite de $f$ by i-position, le produit $f^{'}*g$ est la décalage circulaire vers la droite de $f*g$ par i-position. En effet, la décalage circulaire vers la droite de $f$ par $i$-position est exactement égale à $rot_i(f) = X^i*f\ mod\ (X^N - 1)$ où $X^N*f\ mod\ (X^N - 1) = f$. Donc $f^{'}*g = X^i*f*g\ mod\ (X^N - 1) = X^i*(f*g)\ mod\ (X^N - 1) = rot_i(f*g)$ , Donc:

$$rot_i(f*g) = rot_i(f)*g \qquad (1)$$

Si on additionne $f*g$ avec sa décalage à droite par $i$-position (1), on obtient: $f*g + rot_i(f*g) = f*g + X^i*f*g\ mod\ (X^N - 1) = (f + X^i*f)*g\ mod\ (X^N - 1) = (f + rot_i(f))*g$, puis:

$$f*g + rot_i(f*g) = (f + rot_i(f))*g \qquad (2)$$

Chaque coefficient d'un polynôme de $\Re$ est inférieur à $q$ cela signifie qu'il s'écrit sur $\lceil log_2(q-1) \rceil$ bits. Donc chaque décalage de $r*h\ modulo\ q$ par $s$-bit où $s$ est multiple de $\lceil log_2(q-1) \rceil$, correspond à un nouveau produit $r^{'}*h\ modulo\ q$ où $r^{'}$ est le décalage circulaire vers la droite de $r$ par $s$-bit, et l'addition d'un produit et son décalage $r*h + rot_s(r*h)\ modulo\ q$ correspond à un nouveau produit $r_s*h\ modulo\ q$ où $r_s = r + r^{'}$.

Donc si une étiquette possède dans sa mémoire un produit $r*h\ modulo\ q$, elle peut facilement calculer un nouveau produit $r_s*h\ modulo\ q$ en utilisant des additions et des décalages.

### C.3.3 Le protocole d'authentification

Comme décrit la Figure C.11, sur la base des notations indiquées au Tableau C.5, une fois l'étiquette interrogée par le lecteur, elle génère une valeur aléatoire $M$ en utilisant une fonction de hachage réitérée, et ensuite elle construit un polynôme binaire $m$ (challenge) à partir du

| | |
|---|---|
| $r*h \ (mod\ q)$ | Désigné par $r*h$ |
| $r^{'}*h \ (mod\ q)$ | Désigné par $r^{'}*h$ |
| $\oplus$ | Opérateur OU exclusif |
| $HW(x)$ | Poids de Hamming de $x$ |
| $rot(x,y)$ | Décalage circulaire vers la droite sur $x$ par $HW(y)$ |
| $rot^{-1}(x,y)$ | Décalage circulaire vers la gauche sur $x$ par $HW(y)$ |
| $rot(x,K'y)$ | Décalage circulaire vers la droite sur $x$ par $K$ fois $HW(y)$ |
| $\lceil x \rceil$ | la partie entière par excès de $x$ |
| $H_{k_t}()$ | Code d'authentification de message basé sur le hachage (HMAC) |
| $k_t$ | Clé secrète à long terme |
| $x_{[0,s-1]}$ | Les $s$ bits les moins significatifs de $x$ |

Table C.5: Notations.

$M$, puis elle génère un nouveau produit $r_s*h$ (cf. Section C.3.2) qu'elle utilise pour chiffrer le challenge $m$. Ensuite, l'étiquette envoie au lecteur le message $e_1$ qui est le chiffré de $m$. Après avoir reçu la réponse de l'étiquette, le lecteur déchiffre $e_1$ *modulo* $q$ en utilisant sa clés privée et récupère le challenge $m$, ensuite il calcule et envoie à l'étiquette $e_2 = rot(r_s*h, r_s*h) \oplus (r^{'}*h)$ et $e_3 = H_M(r^{'}*h)$. Si $e_3$ est correcte, l'étiquette met à jour la valeur du $r*h$ en la remplaçant par la nouvelle valeur $r^{'}*h$ calculée par le lecteur. Ensuite, l'étiquette utilise la nouvelle valeur $r^{'}*h$ pour chiffrer le résultat de OU exclusif de son identité $id$ avec le challenge $m$. Le lecteur récupère l'identité de l'étiquette en calculant $id = (e_4 - r^{'}{}_s*h) \oplus m$ et non pas en déchiffrant le message $e_4$ avec la clé privée pour éviter les attaques par rejeu.

| Lecteur/Serveur | | Étiquette |
|---|---|---|
| | $\xrightarrow{hello}$ | trouver $r*h$, $k_t$, $M$, calculer $M \leftarrow H_{k_t}(M)$, $m = B2P(M)$, $r_t = M_{[0,s-1]}$, $r_s*h = r*h + rot(r*h, \lceil log_2(q-1) \rceil' r_t)$ |
| déchiffrer $e_1 \ (mod\ q)$, trouver $m$ | $\xleftarrow{e_1}$ | $e_1 = r_s*h + m$, $r*h \leftarrow rot(r*h, \lceil log_2(q-1) \rceil' r_t)$ |
| choisir $r^{'} \in D_r$, calculer $e_2 = rot(r_s*h, r_s*h) \oplus (r^{'}*h)$ et $e_3 = H_M(r^{'}*h)$ | $\xrightarrow{e_2, e_3}$ | trouver $r^{'}*h = e_2 \oplus rot(r_s*h, r_s*h)$, vérifier $e_3$, calculer $r^{'}{}_s*h = r^{'}*h + rot(r^{'}*h, \lceil log_2(q-1) \rceil' M)$, |
| trouver $id = (e_4 - r^{'}{}_s*h) \oplus m$ | $\xleftarrow{e_4}$ | $e_4 = r^{'}{}_s*h + id \oplus m$, $r*h \leftarrow r^{'}*h$ |

Figure C.11: Le protocole d'authentification proposée pour $p = 2$

### C.3.4 L'analyse de sécurité du protocole

La résistance du protocole proposé aux attaques est directement dérivée des propriétés de NTRU et du HMAC. L'attaquant est supposé se comporter selon le modèle de Dolev-Yao, c'est à dire avoir le plein contrôle sur le canal sans fil pour rejouer, modifier et enregistrer des messages échangés.

- L'attaque par rejeu: chaque session d'authentification est personnalisée avec des valeurs pseudos aléatoires $m$, $r_s{*}h$ et $r^{'}{*}h$ générées par l'étiquette et le lecteur, ce qui rend le protocole résistant aux attaques par rejeu.

- L'attaque de l'homme du milieu: le but de l'attaquant est de passer pour une étiquette ou un lecteur légitime. Cependant, l'attaquant ne peut pas profiter des modifications que nous avons introduites sur NTRU pour le casser, comme le montre la section 3.4.1. Cependant, le nombre des $r_s{*}h$ que l'étiquette peut générer entre deux sessions d'authentification à partir d'une seule valeur $r{*}h$ est limité à $1/2(N + N^2)$, mais le fait que l'étiquette chiffre un challenge pseudo aléatoire, cela empêche l'attaquant de casser le schéma dans un temps raisonnable. Un moyen d'accroître le nombre de $r_s{*}h$ que l'étiquette peut générer entre les sessions d'authentification mutuelles, consiste à stocker plusieurs valeurs de $r{*}h$ sur l'étiquette de telle façon que l'étiquette utilise de façon aléatoire l'un d'eux à chaque tentative d'authentification.

- L'attaque par déni de service: cette attaque consiste à désynchroniser l'étiquette et le lecteur en délivrant à l'étiquette une valeur incorrecte du produit $r^{'}{*}h$ en modifiant le message $e_2$. Cependant, toute modification de ce message sera détecté nécessairement grâce aux HMAC au niveau du message $e_3$ qui authentifie $e_2$ et qui ne peut être calculé que par un lecteur légitime qui possède la clé privée $(f,\ g)$, ce qui rend le protocole résistant aux attaques par dénis de services.

- La protection de la vie privée: la réponse de l'étiquette $e_1$, est différente à chaque tentative d'authentification, ce qui rend le protocole supporte la protection de la vie privée.

### C.3.5 Évaluation des performances

Ce protocole est spécialement conçu pour les systèmes RFID à bas coûts puisque toutes les opérations coûteuses en ressources comme la multiplication et la génération des polynômes, le calcul modulaire, etc. sont réalisées au niveau du lecteur/serveur qui implémente NTRU, ce dernier est considéré parmi les cryptosystèmes les plus légers. Ainsi la charge de calcul de l'étiquette est limitée à $\lceil log_2(q-1) \rceil (N + S/2)$ opérations de décalage, quelques additions simples, calcul de deux valeurs HMAC en utilisant une fonction de hachage légère, et $n+N$ OU exclusif opérations. D'autre part, le lecteur/serveur implémente des opérations simples comme les algorithmes de chiffrement et de déchiffrement de NTRU.

## C.4 Des approches d'identification à divulgation nulle de connaissance

Les protocoles d'authentification les plus utilisées aujourd'hui se basent sur un partage d'une clé secrète (identité) entre deux entités (vérificateur et prouveur) et l'utilisation des algorithmes de chiffrement symétriques ou asymétriques, fonctions de hachages. Donc l'identité est prouvée par l'aspect secret de la clés partagée ce qui la rend critique, parce que si une troisième entité obtient la clé secrète, elle peut passer pour l'entité légitime. La confidentialité de l'information est problématique dans les systèmes larges et distribués comme beaucoup des systèmes RFID. De plus, il n y a pas de garantie que l'entité vérificatrice ne soit pas malveillante.

Ces limites des approches traditionnelles ont conduit à la découverte des protocoles d'identification connus sous le nom "protocoles à divulgation nulle de connaissance" qui ne nécessitent pas un partage de clés secrète c-à-d que le prouveur est capable de convaincre le vérificateur qu'il tient la clés secrète sans révéler une information sur celui-ci.

Ce type de protocoles est très utile pour beaucoup des applications RFID où le système qui peut authentifier l'étiquette ne peut pas être déterminer à l'avance par exemple à cause de la mobilité des étiquettes dans beaucoup des applications c-à-d qu'il n'y a pas de garantie que le vérificateur ne soit pas malveillant. Cependant, les protocoles à divulgation nulle de connaissance ont la réputation d'être très coûteux en termes de ressources ce qui les rend, tel qu'ils sont, inappropriés pour les étiquettes RFID à bas coûts.

Nous proposons deux approches d'identification à divulgation nulle de connaissance basées sur le protocoles GPS et GPS randomisé. Les approches proposées consistent à stocker sur le serveur des valeurs pré-calculées sous la forme des coupons.

L'idée a deux avantages majeurs. Premièrement, les coupons sont stockés sur le serveur et non pas sur les étiquettes, donc l'approche basée sur GPS peut supporter la protection de la vie privée et le nombre de coupons peut être beaucoup plus important que dans les autres approches conduisant ainsi à une plus grande résistance aux attaques par déni de service pour des étiquettes à plus faible coût, et les coupons consommés peuvent être facilement remplacés par de nouveaux. Deuxièmement, l'étiquette supporte des opérations simples des entiers: une ou deux générations nombre pseudo-aléatoires et deux ou trois opérations sur des entiers simples selon la variante (GPS ou GPS randomisé). À ce titre, l'implémentation peut se faire en moins de 1000 GE (Gate Equivalents).

### C.4.1 La RFID et les protocoles à divulgation nulle de connaissance

Plusieurs protocoles d'identification à divulgation nulle de connaissance ont été proposés comme le protocole Fiat-Shamir [17], Guillou-Quisquater [24], Stern [51], Schnorr [48], et GPS [21]. Les mieux connues et les plus efficaces en terme de transmission sont Guillou-Quisquater et Schnorr qui sont basés sur le problème de factorisation et le problème du logarithme discret; respectivement.

Les approches légères à divulgation nulle de connaissance n'ont pas été bien étudiées par les chercheurs, seules quelques approches ont été proposées dans la littérature. À notre connaissance, le plus envisageable parmi toutes les tentatives dans la littérature est l'approche GPS avec coupons [19] dérivée du GPS et qui est basée sur un compromis entre le niveau de la sécurité et la capacité de stockage de l'étiquette. Cependant, pour un niveau modéré de sécurité, la charge de stockage demandées sur l'étiquette est au-delà des capacités des étiquettes RFID à bas coûts. La variante du GPS utilisant la cryptographie à base de courbes elliptiques est illustrée dans la Figure C.12

paramètres: (EC, +) une courbe elliptique
$P$ un point du EC
A, B et S trois entiers tel que $A \gg B.S$
La clés privée: $s \in [0, S[$
La clés publique: $V = -sP$

| **Étiquette** | | **Lecteur/Serveur** |
|---|---|---|
| choisir $r \in_R [0,\ A[$, calculer $X = rP$ | $\xrightarrow{X}$ | choisir $c \in_R [0,\ B[$ |
| vérifier $c \in [0,\ B[$ | $\xleftarrow{c}$ | |
| calculer $y = r_i + s \times c$ | $\xrightarrow{y}$ | vérifier $y \in [0, A + (B-1) \times (S-1)[$ , et $yP + cV = X$ |

Figure C.12: GPS utilisant la cryptographie à base de courbes elliptiques [21]

Cependant, le protocole GPS ne peut pas offrir la vie privée, même si l'étiquette ne renvoie pas sa clé publique au lecteur parce que si l'attaquant interroge le tag en utilisant le même challenge $c$, il peut calculer $yP - X = c(sP)$ qui est une constante caractérisant chaque étiquette. Ce qui permet à un adversaire de tracer l'étiquette. Une variante randomisée qui garantie la protection de la vie privée a été proposée par Bringer et al. [9]. La différence avec le schéma GPS d'origine est que le GPS randomisé nécessite deux multiplications d'un point de la courbe par un scalaire au lieu d'un seul dans le schéma GPS. Le calcul du point supplémentaire $X'$ de la courbe elliptique à partir de la clés publique du vérificateur $U = tP$ assure que seul le

vérificateur qui possède la clé privée $t$ peut faire la vérification comme le montre la Figure C.13.

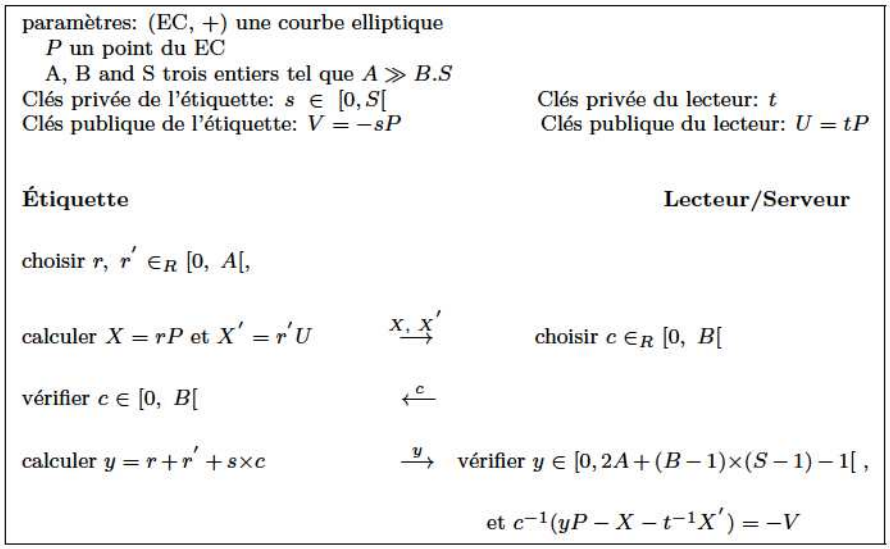


Figure C.13: GPS randomisée utilisant la cryptographie à base de courbes elliptiques [9]

Cependant, GPS et GPS randomisée ne peuvent pas être implémentés sur les étiquettes RFID à bas coûts en raison des importantes ressources requises pour l'opération de multiplication d'un point de la courbe par un scalaire.

### C.4.2 Les approches proposées

Les approches proposées sont basées sur GPS et GPS randomisée spécialement conçus pour les étiquettes RFID à bas coûts. Il s'agit de réaliser un pré-calcul afin de générer des coupons qui seront stockés au niveau de la base de données.
Le principal avantage de nos approches par rapport au GPS avec des coupons est que la variante proposée basée sur GPS peut garantir la protection de la vie privée et le fait que la charge de stockage est supporté par le serveur et pas par l'étiquette. Par conséquent, le nombre de coupons peut être cent fois ou plus le nombre de coupons du GPS avec coupons, tout en réduisant le coût des étiquettes.

Dans nos approches, les étiquettes ont besoin uniquement d'implémenter un générateur de nombre pseudo-aléatoire déterministe (PRNG) avec une graine secrète aléatoire (comme dans le GPS avec des coupons) et des opérations simples sur des entiers (addition et multiplication), de stocker une clé secrète $s$. Notez que pour chaque étiquette, les coupons stockés sur le serveur sont calculés par un tiers de confiance, par exemple, en utilisant le même PRNG que celui de l'étiquette et avec la même graine secrète.

### C.4.2.1 GPS+: l'approche basée sur GPS

Comme le montre la Figure C.15, le lecteur interroge l'étiquette en utilisant un challenge $c$. Après avoir reçu le challenge du lecteur, l'étiquette vérifie que $c \in [0,\ B[$ et régénère une nouvelle valeur pseudo-aléatoire $r_i \in [0,\ A[$ où $0 \leq i \leq n-1$, elle calcule et envoie au lecteur la réponse $y = r_i + s \times c$.

On rappel que l'étiquette utilise un générateur des nombres pseudo aléatoires déterministe.

Une fois la réponse de l'étiquette reçue, le lecteur/serveur cherche dans sa base de données une clés publique $V$ qui satisfit à la fois $yP + cV = (r_i + s \times c)P - c(sP) = r_i P$ et le résultant $r_i P$ soit un coupon associé associé à la clés publique $V$. Sinon, le lecteur interrompt la session concluant que l'étiquette est illégitime.

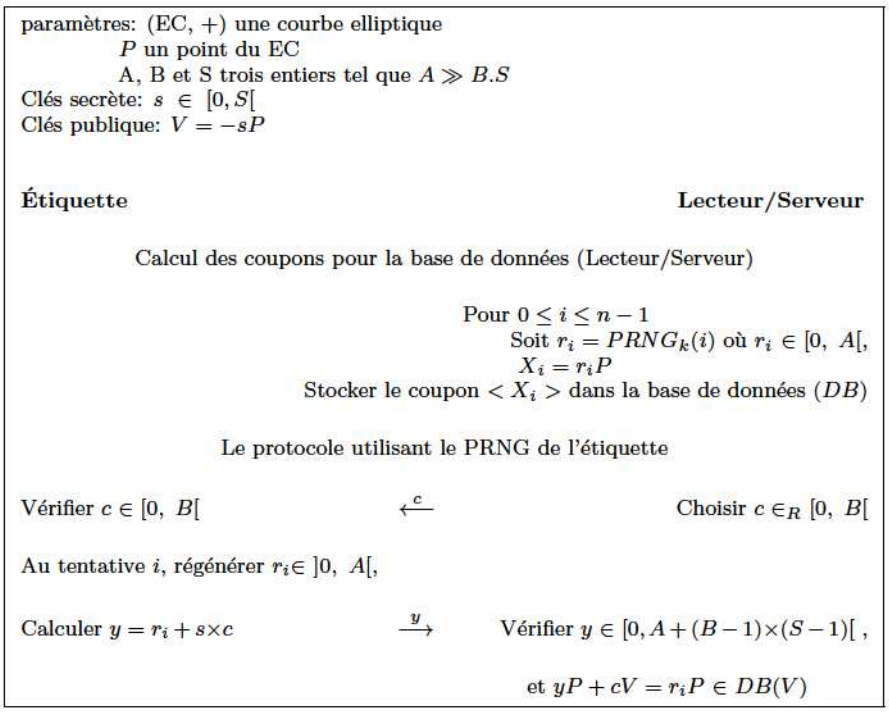


Figure C.14: GPS+: l'approche basée sur GPS

Les spécifications de GPS définissent l'ordre de grandeur de différent paramètres comme suit:

- $s$ est au moins 160 bits de long (i.e. $S \geq 2^{160}$)
- $c$ est de 16, 32, 64 bits de long ( i.e. $B = 2^{16},\ 2^{32}$ ou $2^{64}$)
- $\gg$ signifie "64 ou 80 bits de plus", $r_i$ est de 240, 256 ou 288 bits de long (i.e. $A = 2^{240},\ 2^{256}$ ou $2^{288}$)

Il est à noter que si des challenges avec un faible poids de Hamming sont utilisés, le challenge $c$ peut être plus long, environ 860 bits comme montré dans [38].

### C.4.2.2 GPS++: l'approche basée sur GPS randomisé

Comme le montre la Figure C.15, cette approche est très similaire à l'approche GPS+ et il supporte la protection de la vie privé même si les coupons sont connus à l'adversaire. Cependant, la charge du stockage du serveur est deux fois plus importante que dans GPS+.

Le lecteur interroge l'étiquette en utilisant un challenge $c$. Pour répondre au lecteur, l'étiquette génère deux nombres pseudo aléatoire $r_i$ et $r_i^{'}$ en utilisant son générateur déterministe, et elle calcule et envoie au lecteur la réponse $y = r^{'} + r_i^{'} + s{\times}c$ où $s$ est sa clés secrète. Après avoir reçu la réponse de l'étiquette, le lecteur cherche un coupon $< X,\ X^{'} >$ et vérifie que $-c^{-1}(yP - X - t^{-1}X^{'})$ est une clés publique associé au coupon $< X,\ X^{'} >$ dans la base de données. En effet seul le lecteur légitime qui possède la clés privée $t$ peut faire la vérification suivante:

$$\begin{aligned} -c^{-1}(yP - X - t^{-1}X^{'}) &= -c^{-1}((r_i + r_i^{'} + s{\times}c)P - r_iP - t^{-1}(r_i^{'}tP)) \\ &= -c^{-1}((r_i^{'} + s{\times}c)P - r_i^{'}P) \\ &= -sP = V. \end{aligned}$$

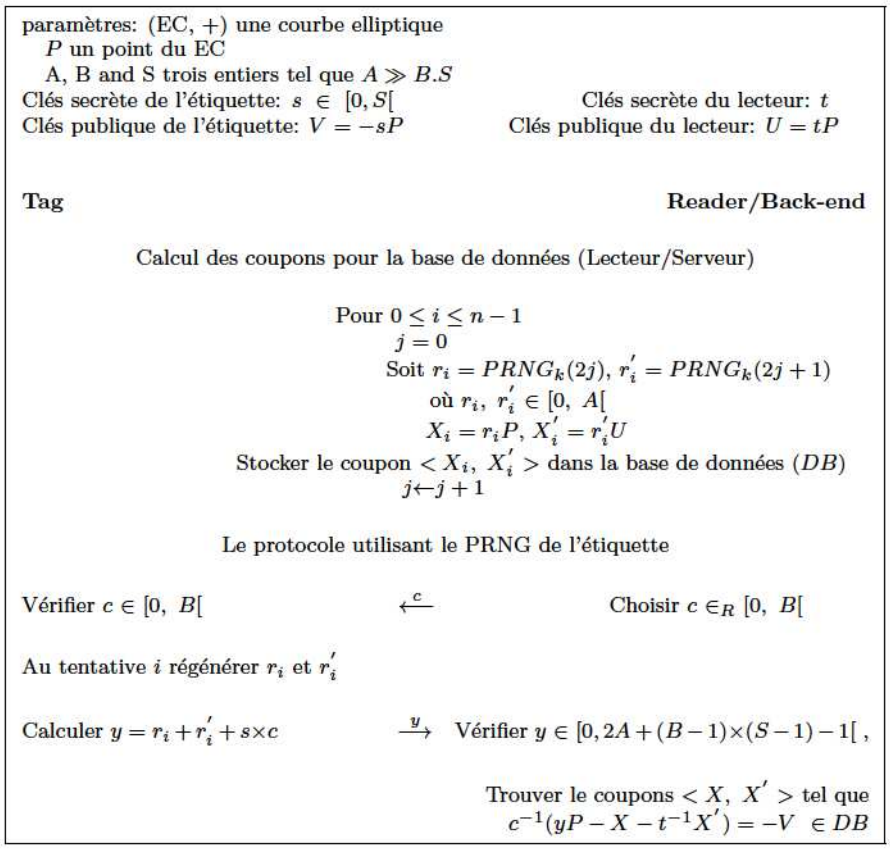


Figure C.15: GPS++: l'approche basée sur GPS randomisé

### C.4.3 L'analyse de sécurité des approches

L'attaquant est supposé se comporter selon le modèle de Dolev-Yao, c'est à dire avoir le plein contrôle sur le canal sans fil pour rejouer, modifier et enregistrer des messages échangés.
Dans les approches proposées, nous supprimons juste le premier message dans GPS et GPS randomisé, donc la sécurité n'est pas affaiblie et la propriété de divulgation nulle est conservée. L'analyse qui suit est valable pour GPS+ et GPS++.

- L'attaque par rejeu: pour chaque session d'identification, lecteur et l'étiquette génèrent de nouvelles valeurs pseudo-aléatoires $c$ et $r_i$, respectivement, ce qui rend chaque session personnalisée. La personnalisation de l'étiquette est de $1/B^l$ où $l$ est le nombre des tours du protocole. Pour garantir un très haut niveau de sécurité, le lecteur peut répéter le protocole $l$ fois de telle façon que $B^l \geq 2^{80}$. Si $B = 2^{32}$, 3 tours du protocole assurent un niveau de sécurité supérieur à 80 bits. Cependant, pour beaucoup des applications RFID, un seul tour est suffisant.

- L'attaque de l'homme du milieu: pour qu'un attaquant soit identifié comme une étiquette légitime, il doit générer un message valide $y'$ ou retrouver la clé secrète $s$ de l'étiquette. Cependant, l'attaquant ne peut pas produire un message valide $y'$ en utilisant les messages échangés auparavant grâce au challenge $c$ généré par le lecteur lui même.
  L'attaquant ne peut pas non plus récupérer la clé secrète $s$ à partir des messages échangés entre l'étiquette et le lecteur parce que le problème du logarithme discret sur les courbes elliptiques est supposé difficile.

- La protection de la vie privée: pour GPS+ on suppose que l'attaquant n'a pas accès aux coupons, si non le protocole ne peut pas supporter la protection de la vie privée comme c'est aussi le cas pour GPS et GPS avec coupons.
  La réponse de l'étiquette dans GPS+ est indistinguable ce qui signifie que GPS+ supporte la protection de la vie privée.
  GPS++ supporte la protection de la vie privée même si les coupons sont publiquement connus, car GPS randomisé garantie un très haut niveau de protection de la vie privée.

### C.4.4 Évaluation des performances

Chacune des approche proposées est composée de deux étapes uniquement. Toutes les opérations coûteuses comme la multiplication scalaire-point, stockage des coupons sont réalisées au niveau

du serveur.
L'étiquette implémente uniquement un générateur déterministe des nombres pseudo-aléatoire et des simples opérations sur des entiers: multiplications et additions.
Comme GPS+ nécessite moins de ressources sur l'étiquette que le GPS avec coupons, il peut être implémentée en moins de 1000 GE en utilisant des challenges avec un faible poids de Hamming, comme le montrent McLoone et al. [39]. L'approche GPS++ nécessite une quantité des ressources sur l'étiquette similaire au GPS+, car il n'a pas besoin de primitives supplémentaires, seulement quelques cycles d'horloge supplémentaires.

## C.5 Conclusions et perspectives

Nous avons présenté un simulateur et trois protocoles d'authentification pour les RFID à bas coûts.

Le simulateur est capable de simuler les différents scénarios dans les systèmes RFID avec des étiquette fixes et mobile et une étude de puissance. Nous avons montré grâce au simulateur simulateur que le cryptographie symétrique est exclu d'être utilisé dans le standard EPC Class-1 Generation-2 ou dans les futur standards scalables pour les systèmes RFID à bas prix lorsque la protection de la vie privée est nécessaire en raison du temps d'identification très court pour les grands nombres des étiquettes. Cela ne donne pas d'autres alternatives que de concevoir des approches légères basées sur la cryptographie à clé publique.

Les trois protocoles d'authentification utilisent la cryptographie à clés publiques. Le premier protocole est basé sur une adaptation que nous avons introduits sur le cryptosystème NTRU. Ce protocole nécessite des opérations simples au niveau des étiquettes et il garantie un très haut niveau de scalabilité (passage à l'échelle) grâce à la légèreté de NTRU.
Les deux derniers protocoles considèrent le problème de divulgation nulle pour les étiquettes à très faible coûts et ils sont basés sur GPS et GPS randomisé. Nous faisons usage de coupons stockés au niveau de la base de données afin de minimiser les ressources dédiées à la cryptographie au niveau des étiquettes. Tout d'abord, nous avons proposé une approche d'identification basée sur GPS. Les avantages de cette approche est que, tout vérificateur qui possède les coupons peut identifier l'étiquette, ce qui rend cette approche ne supporte pas la protection de la vie privée lorsque les coupons sont connus de l'adversaire. Deuxièmement, nous avons proposé une approche d'identification basés sur GPS randomisé, cette approche supporte la protection de

la vie privée, même si les coupons sont connus à l'adversaire, car seule le lecteur légitime qui possède une clé privée peut effectuer la vérification.

Nos perspectives de recherches comprennent:

- l'étude de la possibilité d'utiliser un autre type de la cryptographie à clés publique pour les RFID à bas coûts comme la cryptographie basée sur les codes correcteurs d'erreurs qui est l'une des direction des recherche les plus prometteuse. Il s'agit de développer des techniques et des code qui permettront de minimiser la taille de la clé publique. En effet, la cryptographie basée sur les codes correcteurs d'erreurs nécessite juste des simple opérations logiques, le seul obstacle qui empêche son implémentation sur les étiquettes RFID est la taille importante de la clé publique.

- l'enrichissement du simulateur avec la possibilité d'écrire dans étiquettes, etc.

- la répartition des charges entre plusieurs lecteurs en rendant possible la simulation avec plus qu'un lecteur comme le prévoie le standard afin d'accélérer l'identification ce qui permet de simuler des scénarios plus complexes et plus réalistes..

- l'intégration du simulateur avec des applications RFID existants, ce qui permet de tester de nombreux aspects liés à ces applications sans avoir besoin d'un matériel (étiquettes et lecteurs RFID).

# Bibliography


[1] Consortium for efficient embedded security, efficient embedded security standard 1 version 2. http://grouper.ieee.org.

[2] *EPC TM Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz Version 1.2.0*, 2008. http://www.epcglobalinc.org.

[3] IEEE standard specification for public key cryptographic techniques based on hard problems over lattices. *IEEE Std 1363.1-2008*, pages C1 –69, 10 2009.

[4] Lattice-based polynomial public key establishment algorithm for the financial services industry. *X9.98 standard*, 4 2011.

[5] A.C. Atici, L. Batina, Junfeng Fan, I. Verbauwhede, and S.B.O. Yalcin. Low-cost implementations of ntru for pervasive security. In *Application-Specific Systems, Architectures and Processors, 2008. ASAP 2008.*, pages 79 –84, july 2008.

[6] Arini Balakrishnan, Swetha Krishnan, Cs Advanced, and Instructor Suman Banerjee. Abstract simulation of rfid platform on ns-2.

[7] Lejla Batina, Jorge Guajardo, Tim Kerins, Nele Mentens, Pim Tuyls, and Ingrid Verbauwhede. An Elliptic Curve Processor Suitable For RFID-Tags. Cryptology ePrint Archive, Report 2006/227, 2006.

[8] Lejla Batina, Stefaan Seys, Dave Singelee, and Ingrid Verbauwhede. Hierarchical ECC-Based RFID Authentication Protocol. In *Workshop on RFID Security and Privacy - RFIDSec'11*, Amherst, Massachusetts, USA, June.

[9] Julien Bringer, Hervé Chabanne, and Thomas Icart. Efficient zero-knowledge identification schemes which respect privacy. In *ACM Symposium on Informa-*

*tion, Computer and Communications Security - ASIACCS*, pages 195–205, 2009. http://doi.acm.org/10.1145/1533057.1533086.

[10] Tao Cheng and Li Jin. Analysis and simulation of rfid anti-collision algorithms. In *Advanced Communication Technology, The 9th International Conference on*, volume 1, pages 697–701, feb. 2007. doi ="10.1109/ICACT.2007.358450".

[11] D. Dobkin. *The RF in RFID: Passive UHF RFID in Practice*. Newnes Newton, MA, USA, 2007.

[12] S. Dominikus and M. Aigner. Petra. http://www.iaik.tugraz.at/content/research/rfid/petra/.

[13] Taher El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Proceedings of CRYPTO 84 on Advances in cryptology*, pages 10–18, New York, NY, USA, 1985. Springer-Verlag New York, Inc. http://dl.acm.org/citation.cfm?id=19478.19480.

[14] Kaoutar Elkhiyaoui, Erik-Oliver Blass, and Refik Molva. ROTIV : RFID ownership transfer with issuer verification. In *RFIDSEC 2011, 7th Workshop on RFID Security and Privacy 2011, June 26-28, 2011, Amherst, Massachusetts, USA / Also published in Springer "LCNS", 2012, Volume 7055/2012*, Amherst, United States, month = 06, url = http://www.eurecom.fr/publication/3428, 2011.

[15] EUROPA European Commission. draft recommendation on RFID privacy and security, 2008. http://ec.europa.eu/yourvoice/ipm/ forms/dispatch?form=RFIDRec.

[16] Martin Feldhofer and Johannes Wolkerstorfer. Strong crypto for rfid tags - a comparison of low-power hardware implementations. In *International Symposium on Circuits and Systems - ISCAS*, pages 1839–1842, 2007.

[17] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO*, pages 186–194, 1986.

[18] Christian Floerkemeier and Sanjay E. Sarma. Rfidsim - a physical and logical layer simulation engine for passive rfid. *IEEE Transactions on Automation Science and Engineering*, 6(1):33–43, 2009.

[19] Marc Girault. Low-size coupons for low-cost ic cards. In *Smart Card Research and Advanced Applications, Proceedings of the Fourth Working Conference on Smart Card Research and*

*Advanced Applications, CARDIS 2000, September 20-22, 2000, Bristol, UK*, volume 180 of *IFIP Conference Proceedings*, pages 39–50. Kluwer, 2000.

[20] Marc Girault, Loic Juniot, and Matthew Robshaw. The Feasibility of On-the-Tag Public Key Cryptography. In *Workshop on RFID Security and Privacy -RFIDSec'07*, Malaga, Spain, July 2007.

[21] Marc Girault, Guillaume Poupard, and Jacques Stern. On the fly authentication and signature schemes based on groups of unknown order. *J. Cryptology*, 19(4):463–487, 2006.

[22] Marc Girault and Jacques Stern. On the length of cryptographic hash-values used in identification schemes. In *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*, volume 839 of *Lecture Notes in Computer Science*, pages 202–215. Springer, 1994.

[23] Bill Glover. *RFID Essentials*. O'Reilly Media, Inc, USA, ISBN: 0596009445, february 2006.

[24] L. C. Guillou and J.-J. Quisquater. A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory. In *Lecture Notes in Computer Science on Advances in Cryptology-EUROCRYPT'88*, pages 123–128, New York, NY, USA, 1988. Springer-Verlag New York, Inc.

[25] Jian Guo, Thomas Peyrin, and Axel Poschmann. The photon family of lightweight hash functions. In *Proceedings of the 31st annual conference on Advances in cryptology*, CRYPTO'11, pages 222–239, Berlin, Heidelberg, 2011. Springer-Verlag.

[26] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. Ntru: A ring-based public key cryptosystem. In *Lecture Notes in Computer Science*, pages 267–288. Springer-Verlag, 1998.

[27] Nick Howgrave-Graham, Joseph H. Silverman, and William Whyte. Choosing parameter sets for ntruencrypt with naep and sves-3. CT-RSA'05, pages 118–135, Berlin, Heidelberg, 2005. Springer-Verlag.

[28] Fermentas Inc. IST-1999-12324. final report of european project IST-1999-12324: New european schemes for signatures, integrity, and encryption (NESSIE), April 2004. http://www.cosic.esat.kuleuven.be/nessie/.

[29] International Organization For Standardization ISO. Information processing systems- OSI reference model- part 2: Security architecture, Standard, (7498-2), 1989.

[30] ISO/IEC. International standard ISO/IEC 9798 part 5: Mechanisms using zeroknowledge techniques. December 2004.

[31] Ari Juels and Stephen Weis. Authenticating pervasive devices with human protocols. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO'05*, volume 3126 of *Lecture Notes in Computer Science*, pages 293–308, Santa Barbara, California, USA, August 2005. IACR, Springer.

[32] Elif Bilge Kavun and Tolga Yalcin. A Lightweight Implementation of Keccak Hash Function for Radio-Frequency Identification Applications. In S.B. Ors Yalcin, editor, *RFIDSec'10*, volume 6370 of *Lecture Notes in Computer Science*, pages 258–269, Istanbul, Turkey, June 2010. Springer.

[33] Selim Volkan Kaya, Erkay Savaş, Albert Levi, and Özgür Erçetin. Public key cryptography based privacy preserving multi-context rfid infrastructure. *Ad Hoc Networks Journal*, 7(1):136 – 152, 2009.

[34] Neal Koblitz. Elliptic Curve Cryptosystems. *Mathematics of Computation*, 48(177):203–209, 1987. http://www.jstor.org/stable/2007884.

[35] Yong Ki Lee, Lejla Batina, Dave Singelée, and Ingrid Verbauwhede. Low-Cost Untraceable Authentication Protocols for RFID. In Susanne Wetzel, Cristina Nita-Rotaru, and Frank Stajano, editors, *Third ACM Conference on Wireless Network Security - WiSec'10*, pages 55–64, Hoboken, New Jersey, USA, March 2010. ACM, ACM Press.

[36] Kin Seong Leong, Mun Leng Ng, and P.H. Cole. The reader collision problem in rfid systems. In *International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications (MAPE)*, volume 1, pages 658–661 Vol. 1, aug. 2005.

[37] Chae Hoon Lim and Taekyoung Kwon. Strong and robust rfid authentication enabling perfect ownership transfer. In *Proceedings of the 8th international conference on Information and Communications Security*, ICICS'06, pages 1–20, Berlin, Heidelberg, 2006. Springer-Verlag.

[38] Máire McLoone and Matthew J. B. Robshaw. New architectures for low-cost public key cryptography on rfid tags. In *IEEE International Symposium on Circuits and Systems - ISCAS*, pages 1827–1830, 2007. http://doi.ieeecomputersociety.org/10.1109/ISCAS.2007.378269.

[39] Victor S Miller. Use of elliptic curves in cryptography. In *Lecture notes in computer sciences; 218 on Advances in cryptology—CRYPTO 85*, pages 417–426, New York, NY, USA, 1986. Springer-Verlag New York, Inc. http://dl.acm.org/citation.cfm?id=18262.25413.

[40] David Molnar, Andrea Soppera, and David Wagner. A scalable, delegatable pseudonym protocol enabling ownership transfer of rfid tags. In *Proceedings of the 12th international conference on Selected Areas in Cryptography*, SAC'05, pages 276–290, Berlin, Heidelberg, 2006. Springer-Verlag.

[41] H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problems Control Inform. Theory/Problemy Upravlen. Teor. Inform.*, 15(2):159–166, 1986.

[42] Yossef Oren and Martin Feldhofer. A Low-Resource Public-Key Identification Scheme for RFID Tags and Sensor Nodes. In David A. Basin, Srdjan Capkun, and Wenke Lee, editors, *Proceedings of the 2nd ACM Conference on Wireless Network Security – WiSec'09*, pages 59–68, Zurich, Switzerland, March 2009. ACM, ACM Press.

[43] Roel Peeters and Jens Hermans. Wide strong private RFID identification based on zero-knowledge. Cryptology ePrint Archive, Report 2012/389, 2012.

[44] Axel Poschmann, Matthew J. B. Robshaw, Frank Vater, and Christof Paar. Lightweight cryptography and rfid: Tackling the hidden overheads. In *Information, Security and Cryptology - ICISC*, pages 129–145, 2009.

[45] Jean-Jacques Quisquater, Myriam Quisquater, Muriel Quisquater, Michaël Quisquater, Louis C. Guillou, Marie Annick Guillou, Gaïd Guillou, Anna Guillou, Gwenolé Guillou, Soazig Guillou, and Thomas A. Berson. How to explain zero-knowledge protocols to your children. In *CRYPTO*, pages 628–631, 1989.

[46] M. O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, Cambridge, MA, USA, 1979.

[47] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Journal of Communication ACM*, 21(2):120–126, February 1978. http://doi.acm.org/10.1145/359340.359342.

[48] Claus P. Schnorr. Efficient identification and signatures for smart cards. In *Proceedings on Advances in cryptology*, CRYPTO '89, pages 239–252, New York, NY, USA, 1989. Springer-Verlag New York, Inc.

[49] Tomohiro Sekino, Yang Cui, Kazukuni Kobara, and Hideki Imai. Privacy enhanced rfid using quasi-dyadic fix domain shrinking. In *IEEE Global Telecommunications Conference - GLOBECOM'2010*, pages 1–5, 2010. http://dx.doi.org/10.1109/GLOCOM.2010.5684216.

[50] Adi Shamir. Memory efficient variants of public-key schemes for smart card applications. In *Advances in Cryptology - EUROCRYPT*, pages 445–449, 1994. http://dx.doi.org/10.1007/BFb0053461.

[51] Jacques Stern. A new identification scheme based on syndrome decoding. In *Proceedings of the 13th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '93, pages 13–21, London, UK, UK, 1994. Springer-Verlag. http://dl.acm.org/citation.cfm?id=646758.705701.

[52] Serge Vaudenay. On privacy models for rfid. In *Proceedings of the Advances in Crypotology 13th international conference on Theory and application of cryptology and information security*, ASIACRYPT'07, pages 68–87, Berlin, Heidelberg, 2007. Springer-Verlag. http://dl.acm.org/citation.cfm?id=1781454.1781461.

[53] Elias Weingärtner, Hendrik Vom Lehn, and Klaus Wehrle. A performance comparison of recent network simulators. In *Proceedings of the 2009 IEEE international conference on Communications*, ICC'09, pages 1287–1291, Piscataway, NJ, USA, 2009. IEEE Press. http://dl.acm.org/citation.cfm?id=1817271.1817510.

[54] Mark Weiser. The computer for the 21st century journal. *Scientific American*, 265(3):66–75, January 1991. http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html.