# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

**4,800**
Open access books available

**122,000**
International authors and editors

**135M**
Downloads

Our authors are among the

**154**
Countries delivered to

**TOP 1%**
most cited scientists

**12.2%**
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

## Interested in publishing with us?
## Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

16

# A Knowledge-Based Approach for Detecting Misuses in RFID Systems

Gennaro Della Vecchia[1] and Massimo Esposito[1,2]
[1]*Institute for High Performance Computing and Networking of the National Research Council of Italy*
[2]*University Parthenope*
*Italy*

## 1. Introduction

In the last few years, the Radio Frequency IDentification (RFID) technology has gained increasing attention as an emerging solution for automatically identifying remote objects, people, animals. Early successful applications in asset tracking and supply-chain management and the falling cost of RFID tags have fostered a broadening of the application domain, with new pervasive, RFID-based solutions supporting more user-oriented services. As a result, RFID technology is going to contribute to the massive deployment of sensors in an ever more networked society –a coming Internet of Things where everything is alive, that is, where common objects (including those that are inanimate and abstract) can have individual identities, memory, processing capabilities, along with the ability to communicate and sense, monitor and control their own behavior (Thompson, 2004). In previous works we have explored this technology's potential to facilitate everyday life by seamlessly integrating virtual and physical worlds, varying from personnel tracking and localization to healthcare monitoring (Ciampi et al., 2006; Coronato et al., 2009; Della Vecchia & Esposito, 2010; Esposito et al., 2009; Coronato et al., 2006).

As known, typical RFID systems use a combination of tags, readers and middleware as sketched in Fig. 1. Basically, a reader broadcasts a radio frequency signal to get the data stored on the nearby tags. Data can be a static identification number, user written data or data computed by the tag itself. Having obtained tag data, the reader informs via a wired or wireless network the middleware that in turn stores both tag and reader data in a back-end database.

RFID systems deal with information which very often, if not always, may be critical. Such systems are intrinsically insecure and vulnerable, being prone to threats that can affect tag, reader and middleware as well .

In particular, *tag cloning* is one of the most serious threats to the security of RFID systems. Tag cloning simply consists in catching a tag's unique identifier with the aim of making an exact copy (*clone*) of the cloned tag, so that the clone can pose as the genuine tag, being indistinguishable from the original. Once legitimate tag data are obtained, attackers can reproduce their clone tags on a wide scale and gain access to secured facilities, make fraudulent purchases, alter or even disrupt supply chains, etc.

One conventional approach to secure RFID systems against tag cloning might use cryptographic tags that enable strong tag authentication and make tag cloning a rather

daunting task, but this would skyrocket the cost of the single tag. As a result, a viable solution to defend against tag cloning in RFID systems seems yet to be developed due to the RFID industry's desire to manufacture commercially affordable tags.

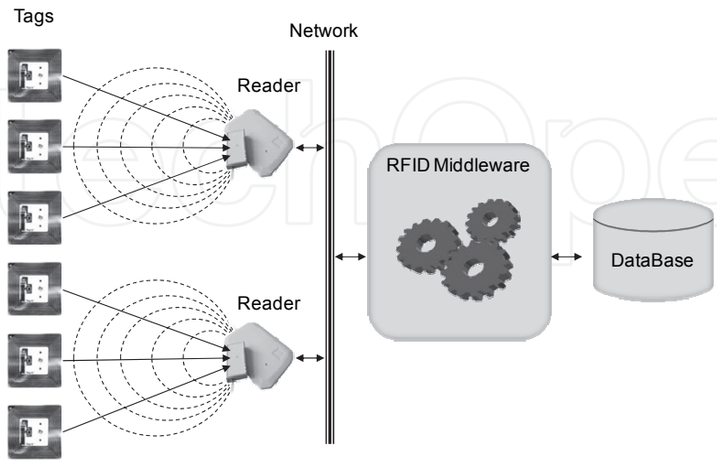



Fig. 1. A typical RFID System Architecture

A less conventional approach to address the tag cloning issue may exploit the well-known security paradigm of "intrusion detection". Generally speaking, an intrusion detection system monitors a given environment and implements a detection method to reveal suspicious activities and respond accordingly. One diffused intrusion detection method is "misuse detection", which utilizes a knowledge base that explicitly models the concept of what is deemed to be "suspicious". Everything that does not match the expected behavior formalized in the knowledge base is considered to be "normal".

In this book's chapter we propose a methodology in which misuse detection uses a knowledge base built upon a "*track & trace*" model relying on the notion of "tag location" to gather all the information required to identify an attack of tag cloning. The knowledge base embedding the track & trace model is formalized in an ontology by means of semantic web languages in order to achieve an unambiguous, well-defined and machine-readable knowledge representation.

The methodology here described stands on three key points: i) the definition of an ontology model formalizing expected and actual profiles, each of them being based on location and tracking information about RFID tagged objects; ii) the application of an inferential engine that, exploiting inference patterns proper to the logic underlying the ontology formalism, checks the consistency between expected and actual profiles of tagged objects; iii) the detection and identification of anomalous conditions in presence of inconsistencies. This methodology led to the design of a misuse detection system aimed at detecting and characterizing tag cloning in RFID applications. Such a system exhibits a reactive and event-dependent behavior in response to new tracking information coming from a network of RFID systems and shows an architecture structured in a set of components operating at middleware layer that can be transparently integrated into existing RFID applications.

The rest of the chapter is organized as follows. Section 2 introduces some preliminary notions, Section 3 discusses motivations and related work, Section 4 describes the proposed methodology and Section 5 illustrates the MDS architecture along with a proof of concept of the knowledge-based approach. Finally, some concluding remarks are reported in Section 6.

## 2. Preliminaries

### 2.1 Intrusion detection taxonomy

Intrusion Detection Systems (IDS) can be classified in several ways. It is common to classify an IDS according to the detection method, the audit source, the usage frequency and the response mechanism (Debar et al., 1999).

Classification by the detection method is the most diffused. Mainly, two kinds of detection methods are considered: misuse detection and anomaly detection.

Misuse detection systems utilize a knowledge base that explicitly models what is not allowed. Everything that does not match the knowledge base is allowed.

Anomaly-based systems, on the contrary, use a model of normal activity and anything that does not match the model of normality is considered an attack. An anomaly detector assumes that all anomalous events are signs of an attack and that all attacks produce anomalous events. Since an anomaly-based system does not model attacks specifically, it can detect previously unknown attacks.

Misuse-based systems, on the other hand, can detect attacks of which they have prior knowledge, being unable to detect new forms of attack but some mutation of those already in the rule base. However, a misuse detection approach paves the way to a clear understanding of the application domain, where users need to be aware of and formalize their knowledge about specific misuse scenarios. This leads to low false positive rates and permits a simple and efficient processing of the audit data.

A different method of classification of IDSs takes into account the type of audit data processed. Three different categories of audit data are common, namely network-based audit data, host-based audit data and application-based audit data.

Network-based sensors collect packets from the protected network in order to perform detection. Some network-based systems use firewall logs as input. These firewall logs contain the headers of the network packets that have been blocked by the firewall. Host-based sensors process audit data generated by a host's operating system. It is very common for this type of sensor to perform detection on the log of system calls that have been executed (Hofmeyr et al., 1998; Kruegel & Robertson, 2004). Application-based sensors process logs created by a user-space application. This kind of sensor is usually used to protect network demons, as several systems exist that process web logs.

IDSs systems can also be classified according to their usage frequency. Online systems operate in real-time and consume audit data as they are generated. This is the most common mode of operation. Other systems are run in offline mode, where the system is activated periodically to look for signs of attack.

Finally, it is also possible to classify Intrusion Detection Systems according to the type of response the system yields when an attack is detected. The most common is passive response, where an attack occurrence is logged or the administrator is alerted by other means (e.g., SMS or email). Active response systems block an incoming attack so that it cannot succeed. They are usually referred to as Intrusion Prevention Systems. Depending on the implementation, an active system could, for instance, send a reset packet to tear down the attacker's connection or update the firewall rules so that the attacker is blocked.

### 2.2 Ontology modeling

Historically, *Ontology* is the philosophical study of the nature of being, existence or reality as such, as well as the basic categories of being and their relations. Traditionally listed as a part of

the major branch of philosophy known as metaphysics, Ontology deals with questions concerning what entities exist or can be said to exist, and how such entities can be grouped, related within a hierarchy, and subdivided according to similarities and differences. By extension, the core meaning of "ontology" within Computer Science is a model for describing the world that consists of a set of types, properties, and relationship types. What ontology has in common in both computer science and philosophy is the representation of entities, ideas, and events, along with their properties and relations, according to a system of categories.

The term "ontology" is currently used to mean "a formal, explicit specification of a shared conceptualization" (Gruber, 1995). In this perspective, "conceptualization" relates to an abstract model that identifies the relevant concepts of a certain domain. "Explicit" means that the type of concepts used and the constraints on their use are explicitly defined. "Formal" means that the ontology should be formalized to be machine understandable and enable intelligent agents to infer new statements from existing ones based on a set of rules. "Shared" means that an ontology captures consensual knowledge of a community. Simply put, ontology refers to a formalization of knowledge in a given domain.

An ontology can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms. In other words, ontology is the concept which is separately identified by domain users, and used in a self-contained way to communicate information. In particular, some of the reasons why someone wants to develop an ontology are to share common understanding of the structure of information among people or software agents, to analyze domain knowledge and enable its reuse, to make domain assumptions explicit, to separate domain knowledge from the operational knowledge.

An ontology structure holds definitions of concepts, binary relationships between concepts and attributes. Three types of relationship may be used between concepts: generalization, association, and aggregation. Relationships may be symmetric, transitive and have an inverse. Concepts, relationship types and attributes and rules, put together, enable the description of a schema in terms of abstraction. On the other hand, concrete objects populate the concepts, concrete values instantiate the attributes of these objects and concrete relationships instantiate relationships.

The semantic web languages used to formalize an ontology are defined with a model-theoretic semantics. In particular, for the language OWL (Web Ontology Language) (Patel-Schneider et al. 2004), a semantics was defined so that very large fragments of the language can be directly expressed using so-called description logics (Baader et al., 2005). Description logics are a decidable subset of first order predicate logic. Namely, OWL DL (where DL stands for "Description Logic") was designed to support the existing description logic business segment and provide a language subset that has desirable computational properties for reasoning systems.

DL-based knowledge bases are built using concept language expressions, and they are usually divided in two distinct parts: intensional and extensional. The intensional part takes the name of T-Box and describes the general conceptual domain model made of concepts and relationships between concepts and attributes, whereas the extensional part is named A-Box and constitutes a (partial) instantiation of the model, since it contains assertions about a set of individuals.

### 2.3 Ontology reasoning

The effective use of ontology modeling in real applications is critically dependent on the provision of efficient reasoning services to support both ontology design and deployment. In

such a direction, DL-based ontologies have stretched the capabilities of DL inference engines, offering a collection of reasoning services implemented through automated reasoning techniques.

The reasoning services provided by a DL inference engine can be classified as basic services, which involve the checking of the truth value for an assertion, and complex services (Donini et al., 1996). Generally speaking, basic services are, for instance, the verification of the subsumption between two concepts or the satisfiability of a concept, whereas complex services implement tasks such as finding all the individuals being in a concept expression, or organizing in form of taxonomy the concept names appearing in a terminology. In more detail, basic services can be classified in services of terminological reasoning and hybrid reasoning, respectively. Terminological reasoning involves only the terminology (i.e. without considering A-Box assertions), whereas hybrid reasoning takes account of both the parts of a knowledge base, i.e., T-Box and A-Box.

On one hand, terminological reasoning services are intended to verify both *Concept Satisfiability* and *Subsumption*, i.e., to check whether a newly defined concept makes sense or is contradictory with respect to the existing T-Box, and to check if a concept C is more general than another concept D, respectively. On the other hand, hybrid reasoning services are aimed at verifying *A-Box Consistency* (with respect to the T-Box) and executing *Instance Checking*. Specifically, A-Box Consistency checks whether a new assertion in the A-Box generates an inconsistency with reference to the T-Box, whereas Instance Checking allow to decide whether an individual is an instance of a concept or not.

The provided complex reasoning tasks vary from system to system, and are defined on top of the basic services above described. The most common are C*lassification* and *Retrieval,* which are terminological and hybrid reasoning services, respectively. Classification consists of explicitly representing the concept taxonomy entailed by the knowledge base, being this taxonomy a graph whose nodes are the concept names appearing in the knowledge base, and the edges represent the subsumption relation between them. This graph can be built by checking the subsumption between every pair of concept names. Retrieval (or *Query Answering*) consists in collecting all the individuals in the knowledge base that are instance of a given concept in every model of the knowledge base.

## 3. Motivations and related work

### 3.1 Motivations

The most challenging security threat in RFID applications is tag cloning. The conventional approach to secure RFID systems against tag cloning is to use cryptographic tags that enable tag authentication and make tag cloning considerable harder. The fundamental difficulties of such an approach revolve around the trade-off between tag cost, level of security, and hardware functionalities. As a matter of fact, RFID tags are typically deployed in great amount and the end-user companies have a strong financial incentive to minimize the tag cost and, thus, the features the tags provide (Lehtonen et al., 2009).

As a result, it is extremely difficult to use cryptography for protecting low cost tags from cloning, due to their limited power, storage and processing resources. Moreover, in order to supply cryptographic components with sufficient power, tags would need to be read from a shorter distance, which would degrade the read-rate of readers (Ranasinghe et al., 2005).

A less conventional approach makes use of location information to detect RFID clone tags. Location-based product authentication is an anti-counterfeiting measure that brand-owners

may use in many situations to fight against product forgery. For instance, in the last years the pharmaceutical industry has been planning to track and trace the history of each single medicine using RFID information as an effective and proactive measure against cloning, instead of using expensive cryptographic tags. Obviously, the goal of this approach is not to make tag cloning harder. Rather, by properly detecting the presence of clone tags and acting accordingly, it aims at nullifying their effects: a tagged product that lacks valid track and trace history can be easily singled out and labeled as not genuine, thus posing a substantial barrier against counterfeit players.

Keeping in mind the aforementioned trade-off between cost and effectiveness, there are at least a couple of good reasons to assume that the location-based approach can be suitably followed when it comes to fight tag cloning. First, efforts focused on the tag itself are intrinsically insecure, because in an RFID system tags constitute the weakest link in the whole chain due to their limited functional capabilities: an attacker, even with poor resources, can violate their security quite easily; on the other hand, it is rather questionable whether it will be ever possible to produce a truly secure RFID tag, able to address all known vulnerabilities without increasing the overall cost. As a second but not secondary consideration, even if some solution may prove itself effective on preventing tag cloning, a really secure RFID system should go beyond prevention measures and provide detection capabilities *when tag cloning has already occurred* (Mirowski & Hartnett, 2007).

All these considerations constitute the rationale which led us to adopt the location-based approach in developing the methodology proposed in this chapter.

### 3.2 Related work

RFID technology raises a number of security and privacy concerns, which may substantially limit its deployment and reduce potential benefits. Among the great deal of papers addressing these concerns, an interesting survey can be found in (Rotter, 2009), with the focus put on the technical aspects of security and privacy.

Most specific literature covers the topic of tag cloning and the efforts made by the research community to tackle this threat through a wide range of solutions.

In costly RFID tags, where resources are less subject to strict constraints, several countermeasures have been devised to combat tag cloning, such as deactivation of tags, encryption, authentication and hash codes (Karygiannis et al., 2007). In (Juels, 2005), some techniques are illustrated for strengthening the resistance of EPC tags against cloning attacks, using PIN-based access to achieve challenge response authentication. In (Weis et al., 2004), the authors proposed a cryptographic approach to lock the tag without storing the access key, but only a hash of the key on the tag instead. The key is stored in a back-end server and can be found using the tag's meta-ID.

Duc et al. (Duc et al., 2006) proposed a communication scheme to protect user privacy in RFID system which is based on a synchronous session key between tags and back-end database server to authenticate each other. A further development of Duc's scheme which overcomes some vulnerabilities has been proposed in (Cheng et al., 2009).

Avoine and Oechslin (Avoine & Oechslin, 2005) proposed another hash-based RFID protocol providing modified identifiers for improving privacy that can be applied for authentication. In addition, hash-based RFID protocols for mutual authentication have been proposed in (Choi et al., 2005; Lee et al., 2006). All these protocols rely on synchronized secrets residing on the tag and back-end server and require a one-way hash function from the tag.

In contrast, in low cost RFID passive tags, due to their small size and strictly constrained resources, complex cryptographic solutions like hash functions cannot be implemented. In (Sarma et al., 2003) the authors mention scarcity of tag resources in low-cost RFID systems as a primary challenge in providing security and privacy mechanisms, and in combating cloning as well. In this perspective, few lightweight authentication protocols that do not require cryptographic hash/keys in the tag have been proposed (Karthikeyan & Nesterenko, 2005; Chien, 2007). Yet another approach to tackle tag cloning uses a Physical Unclonable Function (PUF) (Devadas et al., 2008). PUFs significantly increase physical security by generating volatile secrets that only exist in a digital form when a chip is powered on and running. Its main property is that it is easy to generate but hard to characterize.
As it clearly appears, all these efforts aim at *preventing* tag cloning. However, *detecting* fake tags plays a not lesser role. Once genuine tags have been cloned, the success of a potential attack greatly depends on the capabilities of the whole system to timely recognize it and react accordingly.
Clone detection can be achieved through the gathering of information at the middleware layer. For instance, a very interesting approach has been proposed in (Mirowski & Hartnett, 2007, ib.). The authors have proposed an intrusion detection system for RFID systems, called Deckard, based on a statistical classifier and oriented to the detection of change of tag ownership. This is one of the early researches devoted to the need of intrusion detection systems in RFID. Another remarkable approach, similar to Deckard's intrusion detection architecture, has been proposed in (Thamilarasu & Sridhar, 2008). Its concern goes beyond the change in tag ownership and provides a more generic security framework to detect a variety of RFID malicious attacks.
The methodology we are going to describe in this chapter shares with these latter works the basic concept of applying intrusion detection techniques to identify tag cloning. But, differently, our research efforts have been primarily focused on integrating the principles of ontology modeling and reasoning in the intrusion detection paradigm.
Until recently, few literature can be found about the adoption of the ontology-based approach in developing IDSs. In particular, Raskin et al. (Raskin et al., 2001) advocated the use of ontology modeling in the field of information security in order to provide a common ontology that lets IDS sensors agree on what they observe. Undercoffer et al. (Undercoffer et al., 2003) proposed a target centric ontology for intrusion detection that models properties that are observable and measurable by the target of an attack. Li et al. (Li et al, 2008) described a hierarchical knowledge model to support alert correlation, formalized in an ontology and a set of rules built on top of it. However, to the best of our knowledge, the RFID security literature has not yet addressed applications of inferential engines and ontology modeling to implement intrusion detection techniques in RFID systems, neither system-oriented researches appear to have been developed in that direction.

## 4. Misuse detection system methodology

### 4.1 Track and trace model

The Misuse Detection System (MDS) described in this chapter belongs to the class of IDS characterized by misuse detection as detection method, application-based sensors as type of audit data processed (i.e., the RFID readers), real-time as usage frequency (i.e., audit data are processed as they are generated) and passive response (i.e., an attack occurrence is logged and the administrator is notified of it) (see Sect. 2.1).

The methodology devised to design the MDS relies on a knowledge base containing a "track & trace" model that formalizes all the information required to identify an attack of tag cloning. Such a model essentially relies on the reasoning that *when you know where the genuine tagged object is, the fake/clone ones can be detected*.

More in detail, the model includes static and dynamic profiles to be associated to RFID tagged objects. A static profile is a path composed of points of interest (POI) which a specific tag must visit during its life-cycle under normal working conditions, like for instance those disseminated along a supply chain. A dynamic profile, instead, is a path composed of a tag's actually visited POI, dynamically detected through the supply chain. Dynamic profiles can be built exploiting the set of location events retrieved from a tracing system, such as the EPC network (EPCglobal Inc., 2009). Moreover, the dynamic profile also stores time and type of access (i.e., read/write) of a tagged object visiting a POI.

Both static and dynamic profiles make use of the concepts of *physical location perspective* as opposed to *semantic location perspective*, detailed described in (Coronato et al., 2009, ib.). In principle, a physical location is a precise region identified in terms of proximity to well referenced spots, whereas a semantic location represents the significance of a location within a specific context and may cover more physical locations. In our case, physical locations are the areas covered by RFID readers, while semantic locations can be a country, a city, a building, a room inside a building, a railway station, a watched gate, and so on. A simple typical layout is shown in Fig. 2, where the relationships between physical and semantic perspectives are put into evidence.

The static profile of a tagged object is thus a fixed sequence of semantic locations to be visited through the supply chain, whereas its dynamic profile is the sequence of physical locations actually visited.

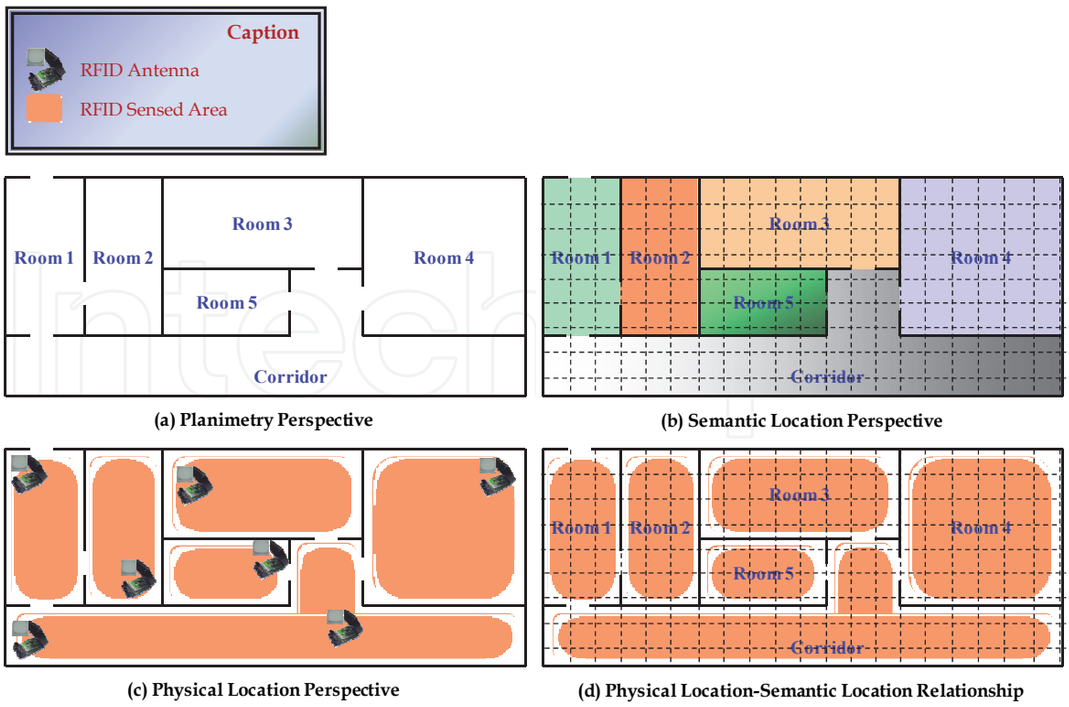



Fig. 2. Representation of the physical and semantic location perspectives

The association between physical location and semantic location is essential to the track & trace model which indeed is based on location-awareness. This means that high-level location information is required, while positioning systems like RFID readers are only able to collect raw location information. In order to fill such a semantic gap, a suitable mapping scheme based on the target supply-chain layout can be established to map physical locations onto semantic ones. This scheme is used in conjunction with the data stored in a tagged object's dynamic profile to identify the semantic locations it in fact passed through.

Physical locations information can be gathered from the audit records generated by RFID read/write operations. As a matter of fact, a typical audit record can be logically structured in <*tagID, readerID, RFIDoperation, timestamp*>, meaning that the *tagID* has been read/written by the *readerID* at the time *timestamp*. The physical location in which *tagID* has been detected is the one associated to *readerID* in the physical location perspective. Then, the mapping scheme established for the target semantic perspective is used to identify the corresponding semantic location. When a tag visits a semantic location, that location becomes a POI for that tag.

As said before, the static profile defined for a given tagged object models its expected behavior within the supply chain, while the dynamic profile stores the ongoing behavior of that object. By comparing static vs. dynamic profiles, the absence of inconsistencies would indicate a "validated" behavior. In contrast, if for any reason the dynamic profile does not satisfy the specifications modeled in the static profile, this fact implies an "abnormal" behavior due to a misuse and –potentially– a threat. In order to effectively detect a real attack, an *a priori* knowledge of possible tag cloning scenarios must be defined (recall that misuse detection techniques rely on a knowledge base that must explicitly model abnormal situations - see Sect. 2.1).

For this reason, in addition to the "normal" static profile associated to each RFID tagged object, a set of "abnormal" static profiles has been formalized with the aim of characterizing known types of clone attack. Once an inconsistency between normal static and dynamic profiles has been revealed for a certain tag, this tag's dynamic profile is checked against the set of abnormal profiles to detect if and what type of attack is going on. Some instances of abnormal static profiles indicating a clone attack are here reported:

- an abnormal profile which includes multiple and non-bordering semantic locations visited at the same time, meaning that a tagged object expected to be at a particular POI has been simultaneously detected in a different semantic location of the supply chain;
- an abnormal profile which includes non-bordering semantic locations visited in a time interval that is inconsistent with the distance between those locations, meaning that a tagged object is moving along the supply chain too fast to be genuine;
- an abnormal profile which includes one or more semantic locations visited by a tagged object not in accordance with the normal static profile, meaning that the object has been detected at one or more POI where it is not expected to be;
- an abnormal profile which includes multiple detections of the same tagged object in a given semantic location, indicating the presence of multiple copies of the same tag. This situation must not be confused with the well-known issue of *collision detections* which affects raw RFID interactions and is usually handled by RFID systems at a lower architectural level;
- an abnormal profile which includes operations performed on a tagged object not allowed under normal working conditions, like for instance too many access to the tag occurring in a fixed time interval at the same semantic location.

More abnormal profiles can be formalized to exhaustively cover all the scenarios and adaptively respond to new kind of attacks as well.

### 4.2 The ontology formalization of the model

The knowledge base at the core of the "track & trace" model has been formalized in an ontology by means of the semantic web languages in order to achieve an unambiguous, well-defined and machine-readable knowledge representation.

In particular, this ontology –called "Track and Trace"- has been devised and implemented in terms of relevant concepts and properties. It is depicted in Fig. 3 as a graph whose nodes represent concepts and sub-concepts while the edges represent properties.

A property can be used to model either binary relationships between concepts or simple attributes. Concepts and properties have been formalized in OWL DL. The choice of this language is due to i) the high degree of expressiveness and modeling power, that enable to formalize complex models in an accurate and sound way; ii) its model-theoretic semantics, that allows to automatically apply reasoning techniques, as discussed in the next section.

For the sake of clarity, we subdivided the ontology in four logical sections. Each property is defined in terms of domain (the set of possible subject concepts) and range (the set of possible object concepts or data types). Besides, the inverse property is reported, if applicable, and the transitiveness is specified, if existing. It is worth noting that each sub-concept inherits super-concept properties and adds new specialized ones.

The first section of the ontology is devoted to model the static profile of a tagged object. In Fig. 3 (a), main concepts and properties are outlined , while the complete list of properties for each concept and sub-concept, not shown in figure for conciseness, is reported in Tab. 1.

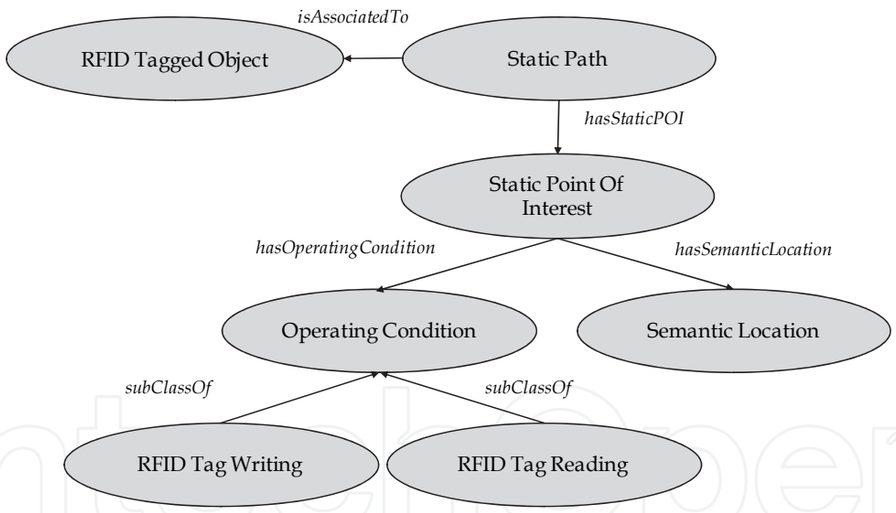



Fig. 3. a) "Track & Trace" Ontology: Static Profile

| Property | Domain | Range | Inverse | Trans. |
|---|---|---|---|---|
| hasStaticPOI | StaticPath | StaticPointOfInterest | isStaticPOIOf | No |
| hasStaticPathID | StaticPath | *Datatype*: String | - | - |
| isAssociatedTo | StaticPath | RFIDTaggedObject | hasStaticPath | No |
| hasOperatingCondition | StaticPointOfInterest | OperatingCondition | isOperatingConditionOf | No |
| hasSemanticLocation | StaticPointOfInterest | SemanticLocation | isSemanticLocationOf | No |
| hasMaxOperationNumber | OperatingModality | *Datatype*: Integer | - | - |
| hasMaxTimestamp | OperatingModality | *Datatype*: Time | - | - |

Table 1. Properties defined for a static profile

The second section of the ontology models a dynamic profile. Main concepts and properties are outlined in Fig. 3 (b), and a detailed list of properties is reported in Tab. 2.

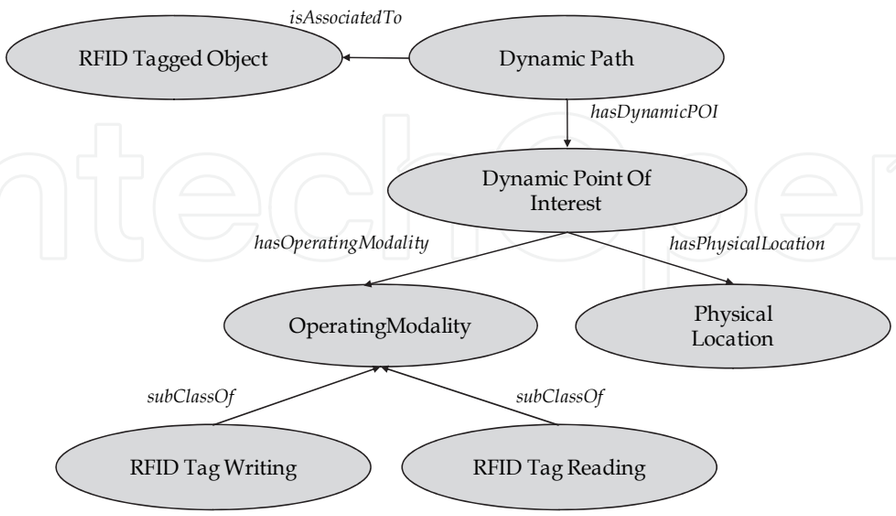



Fig. 3. b) "Track & Trace" Ontology: Dynamic Profile

| Property | Domain | Range | Inverse | Trans. |
|---|---|---|---|---|
| hasDynamicPOI | DynamicPath | DynamicPointOfInterest | isDynamicPOIOf | No |
| hasDynamicPathID | DynamicPath | *Datatype*: String | - | - |
| isAssociatedTo | DynamicPath | RFIDTaggedObject | has DynamicPath | No |
| hasOperatingModality | DynamicPointOfInterest | OperatingModality | isOperatingModalityOf | No |
| hasPhysicalLocation | DynamicPointOfInterest | PhysicalLocation | isPhysicalLocationOf | No |
| hasSemanticLocation | DynamicPointOfInterest | SemanticLocation | isSemanticLocationOf | No |
| hasOperationCounter | OperatingModality | *Datatype*: Integer | - | - |
| hasTimestamp | OperatingModality | *Datatype*: Time | - | - |

Table 2. Properties defined for a dynamic profile

The third section of the ontology is devised to specify location information as outlined in Fig. 3 (c) and detailed in Tab. 3.

| Property | Domain | Range | Inverse | Trans. |
|---|---|---|---|---|
| isPartOf | SemanticLocation | SemanticLocation | hasPart | Yes |
| isBorderingOn | SemanticLocation | SemanticLocation | - | No |
| hasSemanticLocationName | SemanticLocation | *Datatype*: String | - | - |
| hasPhysical LocationID | PhysicalLocation | *Datatype*: String | - | - |
| maps | PhysicalLocation | SemanticLocation | isMappedWith | No |
| covers | RFIDReader | PhysicalLocation | isCoveredBy | No |
| hasReaderID | RFIDReader | *Datatype*: String | - | - |

Table 3. Properties defined for location information

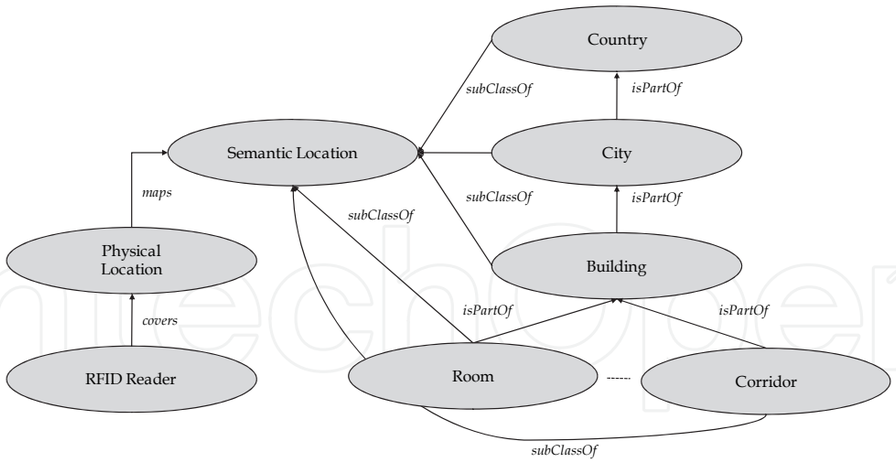



Fig. 3. c) "Track & Trace" Ontology: Location Information

Finally, the last part of the ontology specifies information contained in an audit record in terms of concepts and properties, as outlined in Fig. 3 (d) and reported in Tab. 4.

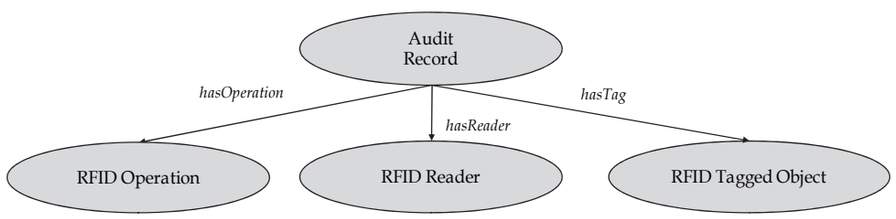



Fig. 3. d) "Track & Trace" Ontology: Audit Record

| Property | Domain | Range | Inverse | Trans. |
|---|---|---|---|---|
| hasOperation | AuditRecord | RFIDOperation | isOperationOf | No |
| hasTag | AuditRecord | RFIDTaggedObject | isTagOf | No |
| hasReader | AuditRecord | RFIDReader | isReaderOf | No |
| hasTimestamp | AuditRecord | *Datatype*: Time | - | - |
| hasTagID | RFIDTaggedObject | *Datatype*: String | - | - |
| hasTagData | RFIDTaggedObject | *Datatype*: String | - | - |
| hasValidID | RFIDTaggedObject | *Datatype*: Boolean | - | - |
| isClone | RFIDTaggedObject | *Datatype*: Boolean | - | - |

Table 4. Properties defined for an audit record

### 4.3 The detection procedure

The application of DL reasoning in the context of intrusion detection requires that a class of attacks be modeled and DL formalism be utilized in such a way that the formulas of the logic can be directly and automatically evaluated. This is where the integration between ontology and DL reasoning plays its role.

In particular, the detection procedure defined for the MDS described in this chapter is a specific application of the description logic on the "Track and Trace" ontology, written in

OWL DL, as previously stated. According to the OWL DL model-theoretic semantics, the detection procedure relies on the application of the hybrid reasoning service "A-Box Consistency" (see Sect. 2.3) to verify whether a dynamic profile of a tagged object is consistent with the corresponding static profile.

In particular, starting from the ontology, T-Box and A-Box have been arranged as follows:

- the T-Box contains closed-form definitions of the static profiles associated to the tagged objects, formulated in terms of concepts, properties and axioms on properties.
- the A-Box contains the individuals (instances of concepts) and the instances of properties. It is populated with the actual information pertaining to dynamic profiles of tagged objects, built in terms of audit records, physical and semantic locations.

The A-Box Consistency check allows to verify whether the terminology of the ontology (i.e., the T-Box) admits the existence of an interpretation that satisfies all the assertions and axioms contained in the A-Box. In the MDS perspective, this functionality has been implemented on the basis of the *tableaux reasoning* (Baader and Sattler, 2001) as provided by the DL inference engine proposed in (Esposito, 2007). This tableaux reasoning searches for an interpretation through a process of completion which starts by constructing an initial completion graph from the A-Box. The nodes in the completion graph intuitively stand for individuals and are associated to their corresponding types. Property-value assertions are represented as directed edges connecting nodes. The reasoning iteratively applies the tableaux expansion rules until a clash (i.e., a contradiction) is detected in the label of a node, or until a clash-free graph is found to which no more rules are applicable. Tableaux reasoning has many advantages: not only it eases the design of provably sound, complete and decidable algorithms, but it is also usually quite efficient at solving many problems that commonly affect real applications.

The detection procedure has been devised and developed on top of the reasoning service for A-Box Consistency, as outlined in Fig. 4 and described in detail as follows.

Through the ontology, the user provides both a high level representation of the terminology and the assertive part to be checked for each tagged object under evaluation. The terminology to be checked consists in a normal static profile stored in the T-Box, whereas the assertive part is represented by the dynamic profile, built on demand with the information coming from the audit records and stored in the A-Box.

The DL inference engine executes the task of verifying the A-Box Consistency and may terminate with the answer "true", indicating that the dynamic profile satisfies the normal static profile, i.e., there exists an interpretation of the assertive part that satisfies the terminology specified in the ontology. In other words, this means that the tagged object under investigation is recognized as genuine. On the contrary, if the inference engine terminates with the answer "false", thus indicating that the dynamic profile is not consistent with the normal profile, this implies that the tagged object is not genuine and is expected to be a clone. In order to determine the reason why the terminology is not satisfied by the assertive part, that is to say why a coherent match between dynamic and normal static profiles has not been found, a set of additional executions are launched by the inference engine.

First, such additional executions require a change of the terminology loaded in the T-Box, i.e., the normal static profile is replaced by one of the abnormal static profiles previously described in section 4.1. Then, the A-Box Consistency task is launched again in order to verify whether the dynamic profile satisfies the abnormal static one loaded in the T-Box.

Additional executions continue until the inference engine returns the answer "true" and, finally, a report is generated describing the kind of abnormal static profile in the T-Box that has been just satisfied by the dynamic profile currently in the A-Box.

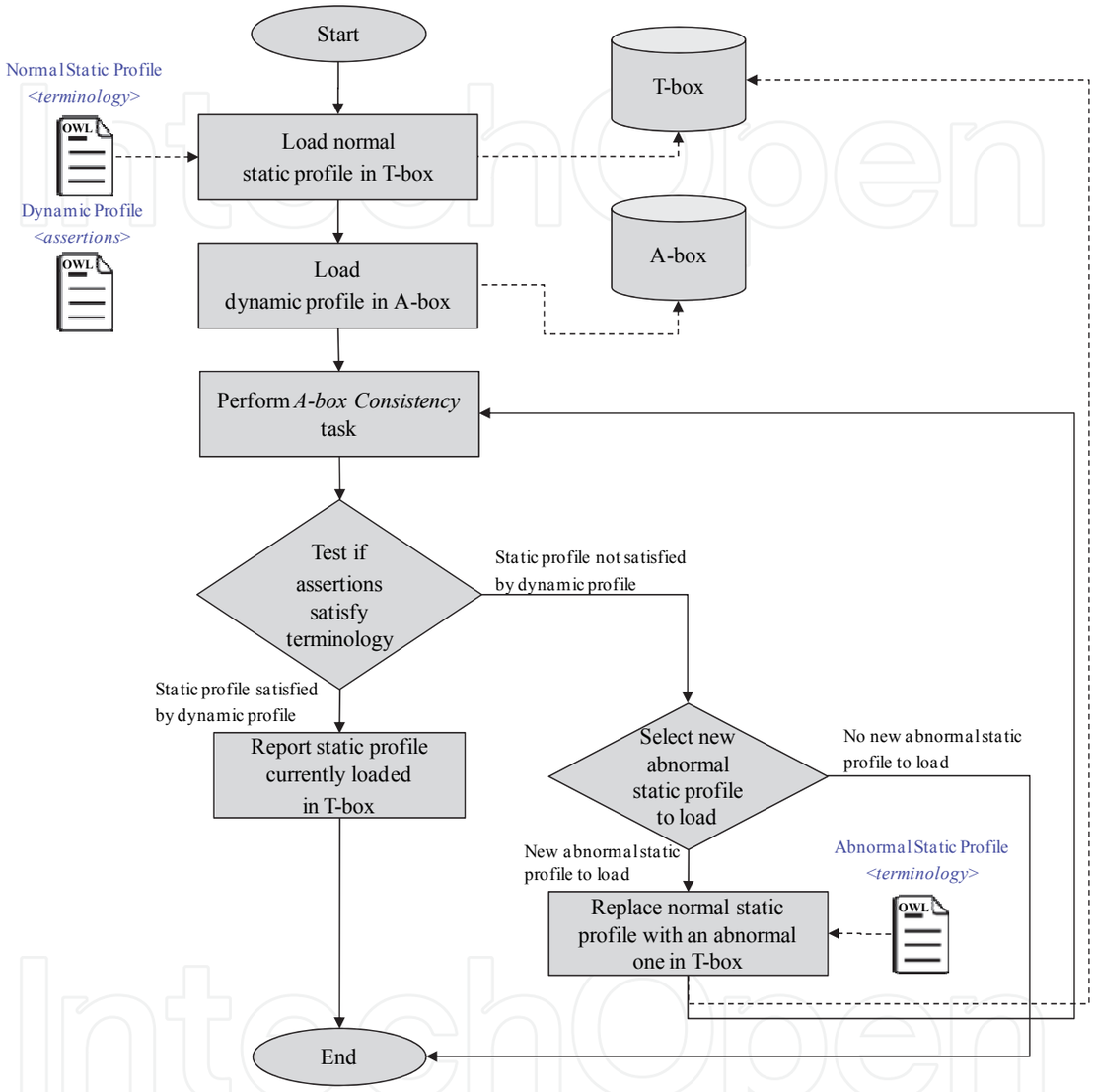



Fig. 4. The detection procedure

In summary, not only can this detection procedure determine the kind of attack, but it also reports a detailed description of how the attack has manifested itself, using the rich and very expressive formalism guaranteed by ontology languages.

## 5. The ontology-based Misuse Detection System

### 5.1 The Misuse Detection System architecture

We designed the architecture of the Misuse Detection System described in this Section on the basis of the methodology illustrated in Sect. 4.

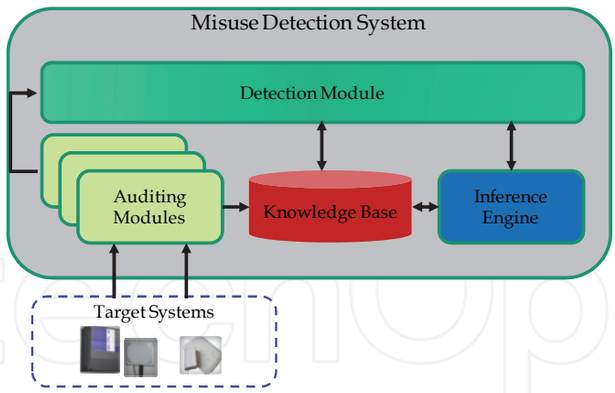



Fig. 5. The MDS architecture

This architecture takes its place at the middleware abstraction layer, on top of the target system layer. A target system is a typical RFID system capable of gathering data that summarize the activity of tags detected in its coverage area. Essentially, a target system produces an audit record for each performed RFID read/write operation.
The MDS architecture is composed of the set of components shown in Fig. 5 and here described:

- *Auditing Modules*. These are the component responsible for collecting the audit records generated by the target systems and storing them into the Knowledge Base. There are as many Auditing Modules as physical locations, each module being associated to a specific RFID reader.
- *Knowledge Base*. It is constituted by the domain knowledge formalized in the ontology described in Sect. 4.2.
- *Inference Engine*. This is the "smart" component with advanced capabilities for performing reasoning services on the ontology stored into the Knowledge Base.
- *Detection Module*. This is the component which executes the core business of the whole MDS. By interacting with the Inference Engine, it is in charge of deciding if and what kind of attack has occurred. The Detection Module exhibits a reactive and event-dependent behavior in response to new generated audit records.

With such components in mind, the MDS operates in the following manner. When a tagged object is sensed by a reader in a Target System, a new audit record is generated with the details of the performed RFID operation (see Sect. 4.1). The corresponding Auditing Module collects this record, stores it in the Knowledge Base and notifies the Detection Module of this event. The Detection Module, in turn, invokes the Inference Engine for processing the data contained in the new audit record. First, the dynamic profile associated to the sensed tagged object is identified. Then, on the basis of the location information stored in the Knowledge Base, the Detection Module looks for the physical location where the tag has been detected, and successively determines the corresponding semantic location.
After that, the dynamic profile is updated in the Knowledge Base with the information pertaining to the visited POI (type of access and timestamp reported in the audit record) or, if the semantic location was never visited before, a new POI and its related information is added to the dynamic profile.
This behavior has been formalized by means of a set of rules conforming to the Event-Control-Action (ECA) architectural pattern. Generally, ECA rules have the form

"on<event>if<condition>then<action>", where <condition> specifies the circumstances that must be verified for the <action> to be carried out whenever an <event> occur. In our specific case, the <event> is the one generated by an Audit Module while <condition> is constituted by the dynamic profile current state. Finally, <action> consists in the updating of the dynamic profile.
Once the Knowledge Base has been updated, the Detection Module launches the detection procedure by invoking the Inference Engine and waits for the answer. Finally, the administrator is notified of the results and the Detection Module is ready to process a new event generated by one of the Auditing Modules.

### 5.2 Proof of concept

In order to give a proof of concept of the knowledge-based approach for detecting misuses in RFID systems presented in this book's chapter, an experimental prototype has been developed at the Institute for High Performance Computing and Networking (ICAR) of the National Research Council of Italy (CNR). This prototype implemented the MDS architectural components as fully portable Java entities on a platform having the following characteristics:

- 2.80GHz Intel® Core™ i7 CPU;
- 8GB of RAM; 240GB of hard disk;
- Windows 7 Professional OS;
- Java SDK 1.6; 1GB of max heap size.

Tests have been performed on this prototype for assessing the response to cloning attacks and the detection accuracy rate. For this purpose, a network of target systems has been simulated through an additional software component in charge of feeding the MDS with suitable devised sets of audit records (see Sect. 4.1) able to reproduce various working conditions.
The simulated scenario consisted of a layout of 50 semantic locations relying on an RFID network made of 50 readers handling up to 20000 tags. For each tag, a static profile has been built with a randomly chosen number of POI between 15 and 50. The minimum time required by a tagged object to move from a POI to another one was set in accordance with the simulated distance among the semantic locations. The maximum number of read/write operations for a single tag allowed at each semantic location was set to 4.
As many as 10 test cases have been designed with the aim of reproducing some combinations of one, more or all the abnormal profiles described in Sect. 4.1 under an increasing traffic of both genuine and clone tags. For each test case, a testing session has been executed on the prototype by injecting the Auditing Modules with the appropriate set of audit records, simulating the traffic of tagged objects in the RFID network. In details, depending on the particular test case, a single attack of clones has been simulated by injecting $n$ copies of a genuine tag with $n$ randomly chosen between 1 and 1000, while multiple attacks have been simulated by injecting $k$ distinct single attacks with $k$ randomly chosen between 2 and 10, for a maximum of 10000 clone tags. Upon completion of a testing session, the resulting data set has been stored for further analysis.

#### 5.2.1 Data analysis

With the aim of assessing the capability of the MDS to respond to cloning attack, a detailed analysis of the data set resulted from the testing activities above has been carried out.

In the following discussion, the term *positive* indicates a clone tag as well as the term *negative* indicates a genuine tag, so that *true positive* (TP) means a clone tag correctly detected whereas *true negative* (TN) means a genuine tag correctly recognized as such. Likewise, *false positive* (FP) stands for a clone tag wrongly treated as a genuine as opposed to *false negative* (FN), i.e., a genuine tag wrongly detected as a clone.
The occurrences of true positives and true negatives have been counted along with the number of false positives and false negatives. Tab. 5 shows the detection performance in terms of accuracy, true positive rate (TPR) and false positive rate (FPR), calculated as follows:

$$ACCURACY = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$TPR = \frac{TP}{TP + FN} \tag{2}$$

$$FPR = \frac{FP}{FP + TN} \tag{3}$$

Furthermore, response times have also been collected, as reported in the last column of Tab. 5.
Looking at the outcome of the data analysis, it can be observed that the experimental Misuse Detection System achieves both high accuracy levels and elevated true positive rates in detecting cloning attacks. Even when these ratings lower as the number and complexity of attacks increase, they still remain quite good, with values of 92.5% of accuracy and 93% of true positive rate for the worst attack scenario considered in the simulation.
The variability of the accuracy rate shown by these experimental results is motivated by the threshold values set respectively for the minimum time required by a tag to move from a POI to another one and for the maximum number of read/write operations for a single tag allowed at each semantic location. Stricter thresholds might yield lower false positive rate but may also let many actual attacks go unnoticed, whereas relaxing the thresholds may increase the true positive rate but also lead to too many false alarms. The actual threshold values were chosen in order to strike a right balance between true positive and false positive rates, with the relief that, in the secure RFID application domain, it is anyhow better to detect false attacks rather than neglect truly dangerous treats.

| Clone tags | Total tags | Accuracy % | TPR % | FPR % | Response Time (min) |
|---|---|---|---|---|---|
| 1000 | 11000 | 100 | 100 | 0 | < 8.0 |
| 2000 | 12000 | 98.33 | 95 | 1 | < 9.0 |
| 3000 | 13000 | 98.46 | 96.66 | 1 | < 10.5 |
| 4000 | 14000 | 97.14 | 97.5 | 3 | < 9.5 |
| 5000 | 15000 | 96.66 | 94 | 2 | < 11.5 |
| 6000 | 16000 | 96.25 | 96.66 | 4 | < 13.0 |
| 7000 | 17000 | 93.52 | 94.28 | 7 | < 14.5 |
| 8000 | 18000 | 94.44 | 93.75 | 5 | < 9.5 |
| 9000 | 19000 | 93.68 | 93.33 | 6 | < 15.0 |
| 10000 | 20000 | 92.5 | 93 | 8 | < 16.5 |

Table 5. Experimental results for the simulated scenario

Moreover, it should be noted that the achieved response times, while affordable, show some non-monotonic trends as the number of tags increases, contrary to what one could expect. This is due to the composite nature of the various test cases devised to stimulate as exhaustively as possible the detection capability of the MDS to respond to cloning attack. In fact, different abnormal profiles require different computational time for reasoning, and it may happen that a more time-consuming profile is checked against a lower traffic of tags in a time shorter than the time required to check a less time-consuming profile against a great deal of tags, or vice versa. However, the overall linearity exhibited by response times –a nearly twofold increase in traffic calls for nearly doubled execution times– indicated a good scalability of the MDS.

## 6. Conclusions

Detection may be seen as the first step in defending against tag cloning and preventing RFID-enabled crime.

As an effort in this direction, the research presented in this book's chapter was intended to investigate whether it is feasible to integrate the principles of ontology modeling and reasoning in the intrusion detection paradigm with the final aim of identifying clone RFID tags. A suitable methodology has been devised for designing a Misuse Detection System which relies on an ontology that explicitly models both normal and anomalous working conditions and uses an inferential engine to dynamically reveal possible inconsistencies in the traffic of RFID tagged objects.

The MDS has an architecture made of a set of components placed at the middleware abstraction layer and exhibits a reactive and event-dependent behavior in response to new tracking information coming from the underlying network of RFID systems. By invoking an inference engine to apply reasoning technique on the formalized knowledge base, the MDS is able to detect cloning attacks and characterize their nature.

An experimental prototype of the MDS has been developed with the purpose of attaining a proof of concept of the knowledge-based approach at the core of this research. Several tests have been carried out on this prototype for assessing its response to various cloning attacks and detection accuracy rate in a simulated scenario. A detailed analysis performed on the experimental data derived from the testing activity showed both high accuracy levels and elevated true positive rates in detecting cloning attacks. Also, the overall linearity exhibited by response times indicated a good scalability of the MDS.

Some concluding remarks can then be drawn.

First, the knowledge-based approach here proposed makes use of semantic-rich models formalized in an ontology which paves the way to a clear understanding of a possible scenario of tag cloning, thus achieving high reliability in the detection process. In addition, the ontology simplicity and intuitiveness can significantly facilitate the tasks of specifying novel attacks of tag cloning, thus keeping the MDS knowledge base up to date.

Second, the basic approach we adopted goes beyond the mere prevention of RFID tag cloning since it allows for the actual detection of clone tags. This is a key point, because when prevention –often very costly to implement– fails, the only way to reduce damages is to react as soon as possible to the presence of clones. Without an automatic mechanism able to timely signal an undergoing cloning attack, it can go unnoticed for days or even weeks. The MDS here proposed constitutes a highly automatic and computationally affordable

solution for processing dynamic RFID audit data and identifying attacks within quite acceptable response times. Moreover, by launching several executions where the terminology to be checked is in turn replaced in accordance with a specific abnormal static profile, it is possible to generate sound reports of how attacks have manifested themselves, each of them being terminologically described with the rich and very expressive formalism proper to the ontology languages .

Third, the MSD has been thought to be transparently and seamlessly integrated into existing RFID systems at the middleware level, with no specific requirements concerning the kind of RFID technology used (passive/active tags, HF/UHF, etc.).

In summary, the outcome of this research seems to be encouraging, suggesting that an actual, scaled-up deployment of the Misuse Detection System here proposed could effectively and proficiently support the detection of clone tags in RFID systems.

Finally, the experimental tests gave a proof of the feasibility of the methodology devised for designing the MDS, which was our main goal. However, an issue remains open as far as MDS's response times are concerned. Improving the performance of the MDS was beyond the initial scope of our research, nonetheless we are confident that the overall execution times might be significantly reduced through a more performance-oriented implementation of the architecture of the MDS, for instance by proceeding with a proper parallelization of its components.

## 7. References

Avoine, G. & Oechslin, P. (2005). A scalable and provably secure hash based RFID protocol, *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 110-114, ISBN 0-7695-2300-5, Kauai Island, Hawaii, March 8-12, 2005.

Baader, F.; Horrocks, I. & Sattler, U. (2005). Description logics as ontology languages for the semantic web, In: *Mechanizing Mathematical Reasoning: Essays in Honor of Siekmann on the Occasion of His 60th Birthday*, *Lecture Notes in Artificial Intelligence 2605,* D. Hutter & W. Stephan, (Eds.), Springer-Verlag, pp. 228-248, , ISBN 3-540-25051-4, New York, NJ, USA.

Baader, F. & Sattler, U. (2001). An overview of tableau algorithms for description logics. *Studia Logica*, Vol. 69, No. 1, pp. 5–40, ISSN 0039-3215.

Cheng, L. M.; So, C.W. & Cheng, L.L. (2009). An Improved Forward Secrecy Protocol for Next Generation EPCGlobal Tag. *Development and Implementation of RFID Technology*, ISBN 978-3-902613-54-7, I-Tech Education and Publishing, Available from: http://www.intechopen.com/articles/show/title/an_improved_forward_secrecy_protocol_for_next_generation_epcglogal_tag.

Chien, H. Y. (2007). SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity, *IEEE Transactions on Dependable and Secure Computing*, Vol. 4, No. 4, pp. 337-340, ISSN 1545-5971.

Choi, E.Y., Lee & S.M., Lee, D.H. (2005). Efficient RFID authentication protocol for ubiquitous computing environment, *Embedded and Ubiquitous Computing, Lecture Notes in Computer Science 3823*, T. Enokido, L. Yan, B. Xiao, D. Kim, Y.S. Dai & L.T. Yang (Eds.), Springer, pp. 945-954, ISBN 3-540-30803-2.

Ciampi, M.; Coronato, A.; De Pietro, G. & Esposito, M. (2006). A Location Service for Pervasive Grids, In: *Advances in Systems, Computing Sciences and Software Engineering*, T. Sobh & K. Elleithy (Eds.), Springer, pp. 119-123, ISBN 978-1-4020-5263-7.

Coronato, A.; Della Vecchia, G. & De Pietro, G. (2006). An RFID-Based Access andLocation Service for Pervasive Grids, In: *Emerging Directions In Embedded And Ubiquitous Computing*, *Lecture Notes in Computer Science 4097*, X. Zhou, O. Sokolsky, L. Yan, E.S. Jung, Z. Shao, Y. Mu, D.C. Lee, D. Kim, Y.S. Jeong & C.Z. Xu (Eds.), Springer, pp. 601-608. ISBN 3-540-36850-7.

Coronato, A.; Esposito, M. & De Pietro, G. (2009). A Multimodal Semantic Location Service for Intelligent Environments: An Application for Smart Hospitals. *Journal of Personal and Ubiquitous Computing*, October 2009, Vol. 13, No. 7, pp. 527-538, ISSN 1617-4909.

Debar, H.; Dacier, M. & Wespi, A. (1999). Towards a taxonomy of intrusion detection systems, *Computer Networks,* April 1999, Vol. 31, No. 8, pp. 805-822, ISSN 1389-1286.

Della Vecchia, G. & Esposito, M. (2010). A Pervasive System for Nuclear Medicine Departments, *Journal of Wireless Personal Communications,* September 2010, Vol. 55, No. 1, pp. 105-120, ISSN 0929-6212.

Devadas, S.; Suh, E.; Paral, S.; Sowell, R; Ziola, T. & Khandelwal, V. (2008). Design and implementation of PUFbased "unclonable" RFID ICs for anti-counterfeiting and security applications, *Proceedings of the 2008 IEEE international conference on RFID*, pp. 58–64, ISBN 978-1-4244-1711-7, Las Vegas, Nevada, USA, April 16-17, 2008.

Donini, F. M.; Lenzerini, M.; Nardi, D. & Schaerf, A. (1996). Reasoning in description logics. In: *Foundation of Knowledge Representation,* G. Brewka, (Ed.), pp. 191-236, CSLI-Publications, ISBN 1-57586-056-2, Stanford, CA, USA.

Duc, D.N.; Park, J.; Lee, H. & Kim, K. (2006). Enhancing Security of EPCglobal Gen-2 RFID Tag against Traceability and Cloning, *Proceedings of the 2006 Symposium on Cryptography and Information Security,* Abstracts pp. 97, Hiroshima, Japan, January 17-20, 2006.

EPCglobal Inc. (2010). EPCglobal Architecture Framework Version 1.4, In: GS1 - The global language of business, 15-12-2010, Available from www.epcglobalinc.org/standards/architecture/.

Esposito, M. (2007). An Ontological and Non-monotonic Rule-based Approach to Label Medical Images, *Proceedings of the 2007 Third International IEEE Conference on Signal-Image Technologies and Internet-Based System,* pp. 603-611, ISBN 978-0-7695-3122-9, Shanghai, China, December 16-18, 2007.

Esposito, M.; Gallo, L.; Coronato, A. & Della Vecchia, G. (2009). An Infrastructure for Pervasive Access to Clinical Data in eHospitals, In: *New Directions in Intelligent Interactive Multimedia Systems and Services, vol. 226/2009 of Studies in Computational Intelligence*, E. Damiani, J. Jeong, R.J. Howlett & L.C. Jain, (Eds.), pp. 431-442, Springer, Berlin/Heidelberg.

Gruber, T. (1995). Towards Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human-Computer Studies*, December 1995, Vol. 43, No. 5-6, pp. 907–928, ISSN 1071-5819.

Hofmeyr, S. A.; Forrest, S. & Somayaji, A. (1998). Intrusion detection using sequences of system calls, *Journal of Computer Security*, August 1998, Vol. 6, No. 3, pp.151-180, ISSN 0926-227X.

Juels, A. (2005). Strengthening EPC tags against cloning, *Proceedings of the 4th ACM workshop on Wireless security*, pp. 67-76, ISBN 1-59593-142-2, Cologne, Germany September 2.

Karthikeyan, S. & Nesterenko, M. (2005). RFID security without extensive cryptography, *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks,* pp. 63-67, ISBN 1-59593-227-5, Alexandria, VA, USA, November 07-10,2005.

Karygiannis, T.; Eydt, B.; Barber, G.; Bunn, L. & Phillips, T. (2007). Guidelines for securing radio frequency identification (RFID) systems, *NIST Special Publication 800-98*, April 2007, available from http://www.nist.gov/manuscript-publication-search.cfm?pub_id=51156.

Kruegel, C. & Robertson, W. (2004). Alert Verification Determining the Success of Intrusion Attempts, *Proceedings of the First Workshop on the Detection of Intrusions and Malware and Vulnerability Assessment*, pp. 1-14, ISBN 3-88579-375-X, Dortmund, Germany, July 6-7, 2004.

Lee, S.; Asano, T. & Kim, K. (2006). RFID Mutual Authentication Scheme based on Synchronized Secret Information, *Proceedings of the 2006 Symposium on Cryptography and Information Security,* Abstracts pp. 98, Hiroshima, Japan, January 17-20, 2006.

Lehtonen, M.; Ostojic, D.; Ilic, A. & Michahelles, F. (2009). Securing RFID Systems by Detecting Tag Cloning, *Proceedings of the 7th International Conference on Pervasive Computing*, pp. 291-308, ISBN 978-3-642-01515-1, Nara, Japan, May 11-14, 2009.

Li, W. & Tian, S. (2009). Preprocessor of Intrusion Alerts Correlation Based on Ontology, *Proceedings of the 2009 WRI International Conference on Communications and Mobile Computing - Volume 03*, pp. 460-464, ISBN 978-0-7695-3501-2, Kunming, Yunnan, China, January 6-8, 2009.

Mirowski, L. & Hartnett, J. (2007). Deckard: A system to detect change of RFID tag ownership, *International Journal of Computer Science and Network Security*, July 2007, Vol. 7, No. 7, pp. 89-98, ISSN 1738-7906.

Patel-Schneider, P.F.; Hayes, P. & Horrocks, I. (2004). OWL Web Ontology Language Semantics and Abstract Syntax, In *W3C Recommendation*, Available from www.w3.org/TR/owl-semantics/.

Ranasinghe, D.C.; Engels, D.W. & Cole, P.H. (2005). Low-Cost RFID Systems: Confronting Security and Privacy, in *Auto-ID Labs Research Workshop*, available from http://www.autoidlabs.org/single-view/dir/article/6/80/page.html.

Raskin, V.; Hempelmann, C.F.; Triezenberg, K.E. & Nirenburg, S. (2001). Ontology in information security: a useful theoretical foundation and methodological tool, *Proceedings of the 2001 workshop on New security paradigms,* pp. 53-59, ISBN 1-58113-457-6, Cloudcroft, New Mexico, USA, September 10-13, 2001.

Rotter, P. (2009). Security and Privacy in RFID Applications, *Development and Implementation of RFID Technology*, ISBN 978-3-902613-54-7, I-Tech Education and Publishing, Available from: http://www.intechopen.com/articles/show/title/security_and_privacy_in_rfid_applications

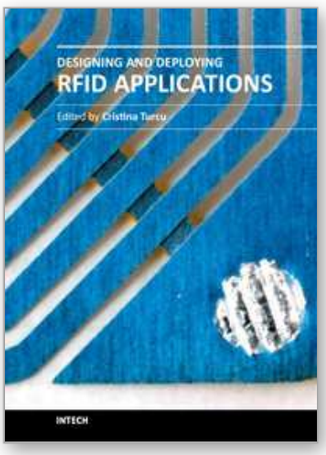
Sarma, S.; Weis, S. & Engels, D. (2003). Radio-frequency identification: Security risks and challenges, *RSA Laboratories Cryptobytes*, Vol. 6, No. 1, (Spring 2003), pp. 2-9.

Thamilarasu, G. & Sridhar, R. (2008). Intrusion detection in RFID systems, *Proceedings of IEEE Military Communications Conference*, pp. 1-7, ISBN 978-4244-2677-5, San Diego, CA, USA, November 17-19, 2008.

Thompson, C. (2004). Everything is Alive. *IEEE Internet Computing*, Vol. 8, No. 1, (Jan-Feb 2004), pp. 83-86, ISSN 1089-7801.

Undercoffer, J. L.; Joshi, A.; Finin, T. & Pinkston, J. (2003). A target-centric ontology for intrusion detection, *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pp. 47-58, Acapulco, Mexico, August 9-15, 2003.

Weis, S.; Sarma, S.; Rivest, R. & Engels, D. (2004). Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems, In: *Security in Pervasive Computing, Lecture Notes in Computer Science 2802*, G. Goos, J. Hartmanis & J. van Leeuwen (Eds.), Springer, pp. 50-59.




**Designing and Deploying RFID Applications**

Edited by Dr. Cristina Turcu

ISBN 978-953-307-265-4
Hard cover, 384 pages
**Publisher** InTech
**Published online** 15, June, 2011
**Published in print edition** June, 2011

Radio Frequency Identification (RFID), a method of remotely storing and receiving data using devices called RFID tags, brings many real business benefits to today world's organizations. Over the years, RFID research has resulted in many concrete achievements and also contributed to the creation of communities that bring scientists and engineers together with users. This book includes valuable research studies of the experienced scientists in the field of RFID, including most recent developments. The book offers new insights, solutions and ideas for the design of efficient RFID architectures and applications. While not pretending to be comprehensive, its wide coverage may be appropriate not only for RFID novices, but also for engineers, researchers, industry personnel, and all possible candidates to produce new and valuable results in RFID domain.



INTECH
open science | open minds

**InTech Europe**
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

**InTech China**
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen