Modeling Security Policies with Recommendations

Nada Essaouini[1], Anas Abou El Kalam[2], Abdellah Ait Ouahman[3]

Cadi Ayyad University, ENSA of Marrakesh – OSCARS Laboratory
Avenue Abdelkarim Khattabi, Guéliz Marrakech, Maroc

Summary
Classical security policies are generally expressed through permissions, prohibitions and obligations. Deontic logic is commonly used for modeling such security rules. We recently emphasize the need of the recommendation modality and we tried to formally specify this new notion by extending the Deontic logic. In this paper we first develop further our Recommendation Specification Language. Then, in order to be able to reason on the security policy and to derive new rules, we give more details about our new recommendation-based axiomatic. Finally, we prove that our new formal system is semantically complete and sound.

Key words:
Information systems security, security policies, access control models, Deontic logic.

1. Introduction

Over the last two decades, information and communication systems have continually been connected, open, pervasive, powerful and complex. At the same time it has become increasingly difficult to achieve an acceptable level of security, especially for sensitive and critical systems. Consequently, several key issues need to be investigated in this priority field: Information systems security.
Dealing with this issue every security study should identify an "organizational security policy": a set of security rules, procedures, or guidelines imposed (or presumed to be imposed) now and/or in the future by an actual or hypothetical organization in the operational environment [1]. Such an organizational security policy usually relies on an access control policy, which defines who has access to what, when and in which conditions. However, a security policy does not guarantee a secure functioning of the system; more important, it could be badly designed, inconsistent or incomplete, especially for huge systems that are congested and under stress. Therefore, The fundamental objective, now is to provide a continuous monitoring of the policy to verify its coherence and detect possible conflicting situations (e.g., situation where a certain user has the recommendation (or the permission) and the prohibition to carry out a certain action on the same object); and also guarantee that all the security objectives are covered by the security mechanisms implementing the policy; etc. In this context, an access control model is often used to rigorously specify and reason on the access control policy (permissions, obligations and prohibitions rules that control the actions performed by subjects on objects).
Moreover, even if most of existing access control models, tools and mechanisms only deal with permissions and prohibitions, it is equally clear that obligations and recommendations become more and more present in current systems, and thus, should be modeled and enforced.
To manage that situation, actually many several works was intended to model obligations. As we know, obligations can be useful to impose some internal or external, manual or automatic actions that should be respected or carried out by users or by the system itself.
So, For example, in the OrBAC model [17], security rules have the form Access Modality (org, r, v, a, c); while Access Modality is a Permission, Obligation or a Prohibition. This rule means: in the context c, organization org grants role r the permission or the obligation or the prohibition to perform activity a on view v.
Bettini et al. distinguish between provisions and obligations [3]. Provisions are conditions that need to be satisfied or actions that must be performed before a decision is rendered, whereas obligations are actions that must be fulfilled by either the users or the system after the decision.
However, these access modalities do not consider the interaction between system and user: system recommend user to do something-user must assume his decision if he refuses the advice of system.
For example, in the General Security Referential document [2] issued by the National Security Agency information systems in France, we find recommendations for the selection and design of cryptographic mechanisms: "It is recommended for any application, to use RSA public exponents strictly greater than 65.536". In this respect, when a user authenticates himself with a RSA public key based certificate (e.g., to access a protected website), it is recommended to use RSA public exponents strictly greater than 65.536.
An other example is about Recommendations on collective cross-border management of copyright and related rights for legitimate on line music services published by the Commission of the European Communities [26]. This document specifies several recommendations like:

Manuscript received November 5, 2011
Manuscript revised November 20, 2011

- *Commercial users should inform collective right managers*
- *of the different features of the services for which they want*
- *to acquire on line rights.*
-
- *where a right-holder has transferred the management of*
- *an on line right to another collective rights manager, without prejudice to other forms of cooperation*
- *among rights managers, all collective rights managers*
- *concerned should ensure that those on line rights are*
- *withdrawn from any existing reciprocal representation*
- *agreement concluded amongst them.*
-

We can cite many other examples, but due to space limitation we can clearly state that recommendations are present in many current and emergent applications.
We recently emphasize the need of recommendations and we considered them as a Deontic logic modality[5]. However, our statements were neither justified nor enough mature. To overcome this limitations, Section 2 of this paper presents our new logical-based framework for modeling recommendations. In particular, we precisely and formally give the syntax, semantic and axiomatization of our Recommendation Language. It is worth to note that this paper is the first one that gives a sounded and complete logical framework with mathematical proofs and justification, which is very important in our context. Afterward, Section 3 shows how using our formalism, for example for querying the security policy and verifying its consistency and coherence. Finally, Section 4 draws conclusions and perspectives.

2. Recommendation Specification Language

A formal language for specifying a security policy should be expressive enough to cover all the requirements of the targeted application. Basically, in order to specify security policies with the different access modalities (permissions, prohibitions, obligations and recommendation), we need first to express norms, e.g., rules which say what must be the case, must not be the case, may be the case or may not be the case. Actually, these norms was already addressed by several logical models such as deontic logic. The latter can be seen as an extension of modal logic that considers modal operators such as obligations, permissions and prohibitions. Note that researches in deontic reasoning within a modal logic point of view has already been done by several works such as by Aqvist [20] and Prior [21]. Moreover, within the context of computer security, several authors like Bieber and Cuppens [22], Glasgow et al. [23], Prakken and Sergot [24], etc. have used deontic logic. In the rest of the following sub-sections, we extend the modal logic in order to model the notions of ”recommendation” and ”inadvisabilities”. We particularly present the axiomatic, semantic and axiomatization as well as all the proofs related to our language.

2.1 Syntax

Let PV be a countable set of propositional variables, with typical members denoted p, q, etc. By means of the Boolean operators ¬ (“not . . .”) and _ (“. . . or . . .”) of classical logic and the modal operator O (“it is obligatory that . . .”) of modal logic, we combine these variables so as to build up the set of formulas of deontic logic given by the rule:

- ::= p | ¬ | (_) | O .

We make use of the standard abbreviations for the other Boolean operators. We supplement the language by the modal operators F, P, and E expressing “it is forbidden that . . .”, “it is permitted that . . .”, and “it is elective that . . .”: F = O¬ , P = ¬O¬ , E = ¬O ¬ . Basically, the specific characteristic of a norm is the consistency of the set of all obligations that make it up. This characteristic corresponds to the formula ¬(O ^ O¬). Seeing that the “obligatory that” is the “forbidden that not” and the “forbidden that” is the “obligatory that not”; this characteristic also corresponds to the formulas:
¬(F ^ F¬) and ¬(O ^ F).
Furthermore, using the equivalences ¬O¬ $ E¬ and ¬F¬ $ P¬ , we can deduce that O ! E¬ and F ! P¬ . The modal operators P (“it is permitted that . . .”) and E (“it is elective that . . .”) keep up similar relations: the “permission that” is the “elective that not” and the “elective that” is the “permission that not”. Hence, we can deduce the following formulas:
O ! P and F ! E .
However, none of the previous modalities is able to directly capture the notion of “recommendation”. Subsequently, we introduce the modal operator R (“it is recommended that . . .”) and we use it to extend the previous set of deontic logic formulas. In fact, let us now consider the set of formulas given by the rule:

- ::= p | ¬ | (_) | O | R .

Let us take a simple example. If we assume that (Read,

Bob, UserGuide) is a formula expressing the fact that Bob reads the user guide, in our language we can express formulas such as R(Read, Bob, UserGuide); meaning that: it is recommended that Bob reads the user guide.

Moreover, to be able to express rules / sentences such as "it is inadvisable that . . .", we supplement the language by the modal operator I: I = R¬ . E.g., the formula I(Execute, Bob, OldVersion) means that executing the old version of the program is inadvisable; i.e., it is recommended to not execute the old version.

In this respect, our new set of formulas allows us to give an account of the consistency of a set of recommendations by means of the formula ¬(R ^ R¬). In fact, seeing that the "recommended that" is the "inadvisable that not" and the "inadvisable that" is the "recommended that not", this formula corresponds to the following formulas: ¬(I ^ I¬) and ¬(R ^ I), it is not possible that something being both recommended and inadvisable.

The question that arises now is: what are the relations between the "obligatory that", the "recommended that" and the "permitted that" on one hand, and the "forbidden that", the "inadvisable that" and the "elective that" , on the other hand. The semantics and the axiomatics of the two next subsections will allow us to show, among others, that the formulas O ! R , R ! P , F ! I and I ! E express indisputable obvious deontic facts.

2.2 Semantics

The most elementary model of obligations is composed of a non-empty set W of states and a relation R on W. Therefore, a deontic frame will be an ordered pair F = (W, R) where W is a nonempty set of states and R is a binary relation on W called accessibility relation: for all states x, the states y such that xRy are those states in which all the obligations in x are satisfied. For this reason, we may also consider that for all states x, the set R(x) = {y: xRy} characterizes the set of all permissions in x. Actually, the formulas of deontic logic are valued at states. The valuation of the formula O at state x depends on the valuation of at states y such that xRy. In this respect, a deontic model is an ordered triple M = (W, R, V) where F = (W, R) is a deontic frame and V is a valuation on W , i.e., a function assigning to each state x in W a subset V(x) of the set PV of all propositional variables. V(x) can thus be considered as the set of propositional variables that x verifies. Subsequently, in the deontic model M , the function V can be extended to the function V defined as follows:

- p 2 V(x) iff p 2 V(x); and ¬ 2 V (x) iff ∉ V(x);
- _ 2 V(x) iff 2 V(x) or 2 V(x);
- O 2 V(x) iff for all states y such that xRy, 2 V(y).

Furthermore, according to the relationships between obligations, permissions, prohibitions, it is a simple matter to check that:

- F 2 V(x) iff for all states y such that xRy, ∉ V(y),
- P 2 V(x) iff for some state y with xRy, 2 V(y),
- E 2 V(x) iff for some state y with xRy, ∉ V(y).

Let us now define the notions of "satisfiability" and "validity" in our model. Let be any formula. We say that is valid in the model M = (W, R, V) iff V(x) for all states x; whereas is said to be valid in the frame F = (W, R) iff is valid in every model M = (W, R, V) based on F. Furthermore, we say that is satisfiable in M = (W, R, V) iff ¬ is not valid in M = (W, R, V); whereas is said to be satisfiable in frame F = (W, R) iff is satisfiable in some model M = (W, R, V) based on F.

Actually, the definitions of satisfiability and validity come from the semantics for modal logic. The reader may easily verify that in all models M = (W, R, V):

- O V (x) iff R(x) {y: V(y)} = R(x), i.e. {y: V(y)} entirely covers R(x)
- P V(x) iff R(x) {y: V(y)} , i.e. {y: V(y)} partially covers R(x).

Seeing that we would like the formulas O R and R P to be valid, the interpretation of the recommendation modal operator R in a model M = (W, R, V) should actually be halfway between the interpretations of O and P, i.e. it should correspond to the following interpretation:

- R V(x) iff {y: V(y)} covers a large part of R(x).

In this respect, the interpretation of I in M = (W, R, V) should correspond to I V(x) iff {y: V(y)} covers a small part of R(x). Note that the notions entirely cover (obligations), partially cover (permissions) and cover a large part (recommendations) perfectly reflect that recommendations are stronger than permissions but not as restricting as obligations. Following our reasoning, we consider that a frame for recommendation is an ordered triple F = (W, R, N) where (W, R) is a deontic frame and N is a neighborhood function on W , i.e. a function assigning to

each state x in W a set N(x) of subsets of R(x). For all states x, we will think of N(x) as the set of large subsets of R(x). Such large subsets will characterize the set of all recommendations in x.

Now, with the recommendation notion, our model is an ordered 4-tuple M = (W, R, N, V) where F = (W, R, N) is a frame for recommendation and V is a valuation on W. In this respect, the function V can be extended (in M) to the function V as follows:

- R V (x) iff R(x) {y: V (y)} N (x).

We will say that a frame F = (W, R, N) is -stable iff for all states x in W , the set N(x) of all large subsets of R(x) is closed for the set-theoretical operation of intersection. We will say that a -stable frame F = (W, R, N) is filtered iff for all states x in W , the set N(x) of all large subsets of R(x) is closed upward, i.e.: for all subsets S, T of R(x), if S is in N(x) and S T then T is in N(x) too.

In the sequel, we always consider that frames of recommendation are fitted out with a serial relation, (for all states x, there exists a state y such that xR y) and with a neighborhood function N such that for all states x, R(x) N(x), N(x), N(x) of all large subsets of R(x) is closed for the set-theoretical operation of intersection and closed upward.

Note that correspondence theory in modal logic teaches us the ways the validity of the modal formula ¬(O O¬) considered above is related to the condition of seriality.

Proof. *Assume that the formula ¬(O O¬) is not valid, then there exists a model M and world x in all possible worlds of M such that M, x ⊭ ¬(O O¬), so (O O¬ V (x). In otherwise, the condition of seriality means that there exists y W such that xR y then V y), therefor ¬ V(y) we conclude that O¬ V(x)*

Let see that the validity of ¬(R R¬) considered above is related to the condition saying that a neighborhood function N is such that for all states x, N (x) of all large subsets of R(x) is closed for the set-theoretical operation of intersection and N (x).

Proof. *Assume that the formula ¬(R R¬) is not valid then there exists a model M = (W, R, V) and world x in all possible worlds of M such that M, x ⊭ ¬(R R ¬), so (R R¬ V(x), then R(x) {y: V(y)} N(x) and R(x) {y: ¬ V(y)} N(x). As N(x) is closed for intersection, R(x) {y: V(y)} {y: ¬ V(y)} N(x), so R(x) {y: (¬) V(y)} N(x), as N(x), we conclude that z W such that (¬) V(y) absurd by definition of V.*

2.3 Axiomatization/completeness

The previous section presents the semantics of our specification and representation language for obligations and recommendations. This is certainly a first step in building a global and robust logical framework; but it remains not sufficient as we need a mean to derive new informations and to reason (e.g., by verification) on our language. Moreover, it seems necessary to give axioms and rules that define the relationships between the different access modalities (obligations, recommendations and permissions) and to give proofs of our axiomatization. To achieve these tasks, we define in this section the axiomatic system LR of our Logic of Recommendation. In addition to the classical axioms of propositional logic, we define the following axioms of LR:

- O( ) (O O),
- O P ,
- O( ) (R R),
- O R ,
- R P .
- R R R( ).
- O( ) (R R)
- R( ) R R

- The axiom O( ) (O O) is called axiom (K). It intuitively corresponds to the fact that the modal operator O is interpreted in models by means of a binary relation.

Proof. *Let x be a world in all possible worlds of model M. We must show that at least one of the following conditions is satisfied:*

(C1) M, x ⊨ O( )

(C2) M, x ⊨ (O O)

Suppose that neither (C1) or (C2) is satisfied in x.

We then have M, x ⊨ O( ), so for all y in W such that xRy, M, y ⊨ (f ® y). And M, x ⊨ Of and M, x ⊭ O (C2 not satisfied), so for all y W such that xRy, M, y ⊨ and there exists at least one y0 such that xR y0 and M, y0 ⊭ then there exists y0 such that xR y0 , M, y0 ⊨ and M, y0 ⊭ we conclude that such y0 satisfies both ( ) , f and Ø wish absurd.

- The axiom O P (axiom D) intuitively corresponds to the fact that in every frame F = (W, R, N), R is such that for all states x, there exists a state y such that xR y.

Proof. *Let x be a world in all possible worlds of model M. We Suppose that M, x ⊨ Of , so for all y such that* xR y we have M, y ⊨ As R is serial then there exists z W such that xR z, so M, z ⊨ then *M, x ⊨ P*

- The axiom O( ) (R R) intuitively corresponds to the fact that the modal operator R is interpreted in models by means of a neighborhood function.

Proof. *Let x be a world in all possible worlds of model* M.

Suppose that *M*, x ⊨ O( ), so for all worlds y such that x*R*y, we have *M*, y ⊨ ( ), wish means that for all y such that x*R*y, we have *M*, y ⊨ iff M, y ⊨ . Then in particular, for the set in *N*(x) contained y such that xRy, we have *M*, y ⊨ iff *M*, y ⊨ . We conclude that M, x ⊨ (R R)

- The axiom O R intuitively corresponds to the fact that for all states x, **R**(x) **N**(x).

Proof. *Let x be a world in all possible worlds of model M . Suppose that M, x ⊨ O so we have for all y such that xRy , M, y ⊨ . As R(x) N(x), then the set of all y such that xRy and M, y ⊨ is in N(x). We conclude that M, x ⊨ R*

- The axiom R P intuitively corresponds to the fact that for all states x, **N(x)**

Proof. *Let x be a world in all possible worlds of model M . Suppose that M, x ⊨ R then the set of all y such that xRy and M y ⊨ is in N(x) and it is not empty as N x Then there exists y such that xRy and M, y ⊨ . We conclude that M, x ⊨ P .*

- The axiom (R R R( ) intuitively corresponds to the fact that for all states x, **N**(x) of all large subsets of R(x) is closed for the set-theoretical operation of intersection .

Proof. *Let x be a world in all possible worlds of model* M. We Suppose that M, x ⊨ (R *R*), so the set of all y in W such that xR y and M, y ⊨ , is in N(x) and the set of all y in W such that xR y and M, y ⊨ is in N x As N(x) is closed for the set-theoretical operation of intersection, then the set of all y in W such that xR y and M, y ⊨ and M, y ⊨ is in N x

Then the set of all y in W such that xR y and M, y ⊨ (is in N x We conclude that M, x ⊨ R( ) .

- The axiom O( ) (R R) intuitively corresponds to the fact that for all states x, **N**(x) of all large subsets of **R**(x) is closed upward.

Proof. *Let x be a world in all possible worlds of model.* We Suppose that *M*, x ⊨ O ( ) and M, x ⊨ R

and show that M, x ⊨ R .

M, x ⊨ R means that the set of all y such that x*R* y and M, y ⊨ is in N(x). *And M, x ⊨ O( ) means that for all y such that xR y, M, y ⊨ ( ), then the set of y such that xR y and M, y ⊨ is included in the set of y such that xR y and M, y ⊨ s N(x) is closed upward, we deduce that the set of y such that xR y and M, y ⊨ is in N(x) wish means M, x ⊨ R*

- The axiom R( ) R R intuitively corresponds to the fact that for all states x, **N**(x) of all large subsets of **R**(x) is closed upward.

Proof. *Let x be a world in all possible worlds of model.* We Suppose that M, x ⊨ R( ), so the set of y such that xRy and M, y ⊨ ( ) is in N(x). Then the intersection of the set of y such that xRy and M, y ⊨ and the set of y such that xR y and M, y ⊨ is in N (x), this intersection is included both in the set of y such that x R y and M, y ⊨ and in the set of y such that xRy and M, y ⊨

As the frame F = (W, R, N) -stable is filtred, we deduce that the set of y such that xR y and M, y ⊨ is in N (x) and the set of y such that xR y and M, y ⊨ is in N (x). We deduce that M, x ⊨ (R R).

Besides that, in addition to the classical inference rules of propositional logic, the inference rules of LR is: “from , infer O ”.

Now, let us give proofs of the most important formulas derivable from axioms and inference rules of LR. Each line of proof is either an instance of an axiom schema, or the application of the inference necessitation rule or the inference Modus Ponens rule

- O Oψ

- O( ),

$$\frac{(A \wedge B) \rightarrow C}{(OA \wedge OB) \rightarrow OC}$$

Proof. *First show that*

A B C
A (B C) (propositional language)
O(A (B C)) (LR inference rule)
OA O(B C) (axiom K)
OA (OB OC) (axiom K)
OA OB OC (propositional language)

Consequently, from , we can deduce
O Oψ
O( ) by applying the inference rule deduced above.

- O Rψ R( ),

$$\frac{(A \wedge B) \rightarrow C}{(OA \wedge RB) \rightarrow RC}$$

Proof. *First show that:*

A B C
A (B C) (propositional language)
O(A (B C)) (LR inference rule)
OA O(B C) (axiom K)
OA (RB RC) (LR axiom)
OA RB RC (propositional language)

As conclusion from *we can deduce*
O Rψ R(ψ) by applying the inference rule demonstrated above.

- O P P( ).

Proof. *First show that O(B C) PB PC*

O(¬C ¬B) ¬C ¬B) (axiom K)

O(B C) ¬PC ¬PB) (PA = ¬O ¬ A by definition)
O(B C) PB PC)

$$\frac{(A \wedge B) \rightarrow C}{(OA \wedge PB) \rightarrow PC}$$

Let show that

A B C
A (B C) (propositional language)
O(A (B C)) (LR inference rule)
OA O(B C) (axiom K)
OA (PB PC) (The formula just demonstrated above)
OA PB PC (propositional language)

As conclusion from we can deduce
O Pψ P(ψ) by applying the inference rule demonstrated above.

Let us now proof that our logical system is sound and complete.

Proposition 1. (soundness property) All formulas derivable from the axioms and inference rules of LR are valid in all frames.

Proof. *It is actually sufficient to prove that the axioms are tautologies and the inference rules are valid. Afterwards, we can use proof by induction on the length of the derivation of in LR to proof that "if is derivable in LR then is valid in all frames'".*
Note that we have already shown that the axioms are valid. Now, let us see that the rule " infers O " is also valid. Let x be a world in all possible worlds of model M. If M, x ⊨ , then for all y W such that xRy, M, y ⊨ . By definition of the validity of the obligation, we can deduce that M, x ⊨ O and then ⊨ O

Proposition 2. (completeness property)
All formulas valid in all frames are derivable from the axioms and inference rules of LR.

Proof. The proof is done by means of a canonical model construction. Let **M** = (**W**, **R**, **N**, **V**) be the model defined as follows:

- **W** is the set of all maximal LR-consistent sets of formulas,
- **R** is the binary relation on **W** , for all x, y in W , xRy iff { : O x} y,
- **N** is the neighborhood function such that for all x in W and for all subsets S of **R**(x), S is in **N**(x) iff there exists a formula such that R x and S = {y **W** : x Ry and y},
- **V** is the valuation function such that for all x in **W** , **V**(x) = {p: p x}.

Let prove that in our canonical model, **R** is serial, for all states x in **W** , **R**(x) **N**(x), **N**(x), **N**(x) is closed for intersection and closed upward.

- **R is serial.**

First show that any set of consisting formulas is contained in some set of formulas maximum-consistent.
Let be a set of consistent formulas. Consider an enumeration, starting with 1, of formulas of the language. We denote the k-th formula according to this enumeration, Ak. We construct a series (k)k N of set of formulas as follows:

1. =
2. k+1 = k {Ak+1 } if k {A k+1} is consistent, and k+1 = k { ¬A k+1 } if not.
k, if {Ak+1} is not consistent, then k { ¬A k+1 }is as well.
Consequently, k, if k is consistent, then k+1 is as well.
As = is consistent hypothetically, all the k are consistent.
Let G the union of all k.
= G by construction, G is consistent (if it is not the case, one of k should be inconsistent) and G is maximal by construction.

Now let prove that R is serial.
Let x W
If , ¬ { : O x}, then O , O ¬ x
As x is LR-consistent, we conclude that { : O x} is LR-consistent tow. So there exists y LR-consistent maximal such that { : O x} y

- **R(x) N (x)**

Let x W, x is not empty, so there exists x, and then O x wish means that for all y W such that xRy, y, then R(x) = {y W such that x Ry, y }, therefor there exists such that O x and R(x) = {y W such that x Ry, y} and then there exists such that R x and R(x) = {y W such that x Ry, y } as x is LR-consistent maximal and O R We conclude that R(x) N(x).

- **N(x)**

We assume that N(x), then there exists such that R x and {y such that xRy and y} = so there exists such that R x and for all y such that
xRy, y or R(x) = Or R is serial, so there exists such that R x and for all y such that x Ry, ¬ y. We conclude that there exists such that R x and O ¬ x, absurd because x is LR-consistent.

- **N (x) is closed for intersection.**

First show that: (R R) x R( ) x
We suppose that R x, R x and R( ) x
As x is LR maximal, we have ¬R( ) x
we have R R R( ), so R( ) belongs to the closure of x, which is absurd because x is consistent.

Let now see that N(x) is closed for intersection.
Let S1 , S2 N(x), then there exists 1 such that R 1 x and S1 = {y W, x Ry and y} and there exists such that R x and S2 = {y W, x Ry and y}, so there exists , such that R R x and S S = {y W, x Ry and y}, then there exists , such that R(x and S S = {y W, x Ry and y}. We conclude that S1 S2 N(x).

- **N (x) is closed upward.**

Let S, T R(x) such that S T and S N(x).
S N(x) then there exists such that R x and {y : x Ry, y} N(x).
Suppose that T N(x), then there exists z T such that xR z and ¬ z as z is LR maximal.
As infer O and z is LR maximal consistent, O(¬) z, then O(¬) V(z), so for all y W such that z Ry, ¬ V(y), and then for all y W such that zRy, ¬ y.
If R x then O(R) x and then for all y such that xRy, R y, so R z, then R V(z), so there exists y0 such that zRy0 and y0.
Absurd because y0 is LR consistent. We deduce that T N(x) and then N(x) is closed upward.

- **Let now see that, by using a proof by induction on the complexity of the formula , for all states x W, x iff V(x)**

Let a formula of length 0 (atomic)
We have x iff V(x).
We suppose that x iff V(x) is true for all formula of length smaller or equal than a given number n and let a formula of length n+1. We have the following cases:

1. *has the form ¬ with is of length n*
2. *has the form with and are of length smaller or equal than n*
3. *has the form O with is of length n*
4. *has the form R with is of length n*

The first two cases are obvious by definition of V
For the third case:
O x means that for all y such that xRy, y, then for all y such that xRy, V(y) and then O V(x).
we suppose that O x and O V(x).
so ¬O x and and for all y such that xRy, V(y).
So P ¬ x and for all y such that xRy, y. Then there exists z such that xRz and ¬ z and z.
absurd because z is LR-consistent. So O x iff O V(x)
For the last case:

R x iff { y such that xRy, y} N(x)
iff {y such that xRy, V (y) } N(x)
iff R V(x).
As a result, if is a formula not derivable in LR (does not belong to the closure of ¬), then ¬ is LR-consistent and there is x W such that ¬ x. Therefore, x and V(x). It follows that is not valid in all frames.

3. Using our formalism

3.1 Specification of the security policy

The axiomatic system defined in the last section, coupled with classical logic axioms could be used for several aims. In this section, two of the possible uses are explained:

(1) query a given policy in order to know which rules apply to a given situation; and

(2) Check the security policy consistency.

To achieve these tasks, it is first necessary to specify the operational rules, the security policy, and the security objectives. In our view, operational rules can be described by means of the propositional logic operators (non modal). For example, to specify that users play roles in their organizations, we can introduce the play predicate between the constant symbols: organizations, users and roles. An instance of this predicate could be for instance Play(Hospital1, Alice, Physician).
Besides that, we suggest expressing security objectives by using modal operators. For example, the R(Nurse, Read, notice) security objective means that it is recommended that the Nurse reads the notice. Finally, we propose expressing security rules using modal formula with at least a non-modal clause (e.g., f Rq). It describes the link between the permissions, prohibitions, obligations, or recommendations and the state of the system. For example, the security rule: “if the patient is minor, it is inadvisable that he/she read its medical file” can be specified by: Age(p) 18 I(p, read, MedicalFile(p). In this rule we have considered that p is a variable of type ”‘patient”’; Age (resp. MedicalFile) is a function that returns the age (resp. the medical file) of a certain patient).

3.2 Querying the security policy

Once we have specified the operational rules, the security policy, and the security objectives of the studied application, we can use our axiomatic to develop a tool which enables a user to query the security policy. For instance, let us assume that security administrator wants to know who is recommended to read a notice? This query is translated in the following logical formula: “ n, Notice(n) R(x, Read, n)”.
Note that there are two ways to program this formula in logical-based languages such as PROLOG. The first one lists the persons who are actually recommended to read a notice; while the second method answers by a formula which corresponds to a sufficient condition that satisfies the query. This second technique of query answering is called intentional answer in [25].

3.3 Checking the security policy consistency

Different techniques can be used to check the security policy consistency, in particular, we can use:

- Axiom-based methods, called Frege-Hilbert methods. The idea is to derive new rules by applying the inference rules to the set of axioms until demonstrating the intended property.
- Natural deduction methods: these techniques are closed to the reasoning used by mathematicians to demonstrate their theorems. In this kind of calculus, every derivation starts by some hypothesis and assumptions.

In our context, it is important to choose the method that (1) gives enough information about the reasons of success or failure while demonstrating a certain security property, (2) identifies the system state that is responsible (3) identifies some resident vulnerabilities in the system or a certain weakness in the security policy specification. This will greatly enhance the system security and rigorously help to refine the security objectives. For these reasons, we suggest using a constructive verification technique such as the ”Tableau method” or its variant ”Gentzen sequence calculus”. In order to prove a certain formula , the main idea is to assume that ¬ is true and to derive a contradiction by successively splitting up ¬ in each of its derived sub-formulas, until obtaining a state satisfying a formula and its negation.

Actually, in this method, we draw a graph where the initial node contains an initial secure state (e.g., a state where certain security objectives are true/satisfied). Then, we progressively apply some derivation rules (specific to this method). At each state we also apply one of the security rules (rules that specify how the system can, must or should evolve). The demonstration is ended when attending a non-secure state (a state where a contradiction is detected). The "Tableau method" can also be used to detect conflicting situations, e.g., if, from a secure state, and by applying the security rules as well as the derivation rules, we reach a state where a certain user has the permission/obligation/recommendation and the prohibition to carry out a certain action on the same object); This problem comes to draw our graph and to look for nodes where one of the following formulas are true: Rp Fq or Pp Fq or Op Fq or Ip Fq or Ip Rq or Ip Oq.

4. Conclusion

A security policy specifies, usually in a textual form, who has access to what, when and in which conditions? Nevertheless, the security policy does not guarantee a secure and correct functioning of the system. Consequently, it is important to associate a model to: abstract the policy and handle its complexity; represent the secure states of a system as well as the way in which the system may evolve; verify the coherence of the security policy and detect possible conflicting situations; guarantee that all the security objectives are covered by the security mechanisms implementing the policy; etc. Deontic logic is a good candidate to model several security properties and modalities. However, none of the existing works have studied the recommendation and inadvisable access modalities, while these concepts are unavoidable in many applications. Several regulations are in fact in the form of recommendations and directives, and these regulations should be reflected in security policies. Modeling recommendations is thus a new challenge in the security policies and models field.

In this paper we first develop further our Recommendation Specification Language. Then, in order to be able to reason on the security policy and to derive new rules, we give more details about our new recommendation-based axiomatic. Finally, we prove that our new formal system is semantically complete and sound. Currently, we are developing mechanisms to integrate our recommendation access modality in existing tools and languages such as Prolog. We also hope extending this work to distributed systems by considering several authorities claiming different kind of statements. Finally, we also expect applying our work to a representative case study.

References

[1] Common Criteria for Information Technology Security Evaluation, Version 3.1 Revision 1, "Part 1: Introduction and general model", CCMB-2006-09-001, 86 p., September 2006.

[2] Référentiel Général de Sécurité version 1.0, annex b1, Mécanismes Cryptographiques, règles et recommandations concernant le choix et le dimensionnement des m ? canismes cryptographiques, version 1.20, Janvier 2010.

[3] C. Bettini, S. Jajodia, X. S. Wang et D. Wijesekera, "Obligation Monitoring in Policy Management", International Workshop, Policies for Distributed Systems and Networks (Policy), Monterey, California, 5-7 June 2002, IEEE Computer Society Press, pp. 2-12.

[4] N. Demeanor, N. Delay, E. Lupus, M. Sloan. "The Ponder Policy Specification Language", International Workshop Policy, Bristol, UK, IEE Computer Society Press, pp.18- 38, 2001.

[5] A. Abou El Kalam and P.Balbiani. A Policy Language For Modelling Recommandations. IFIP Advances in Information and Communication Technology, 297(ISBN 978-3-642-01243-3):176, 2009.

[6] Q. Ni, E. Bertino, J. Lobo, "An Obligation model bridging access control policies and privacy policies", 13th ACM SACMAT), Estes Park, CO, USA, June 11-13, 2008.

[7] M. Hilty, A. Pretschner, D. Basin, C. Schaefer and T. Walter, "A Policy Language for Distributed Usage Control", 12th European Symposium On Research In Computer Security (ESORICS), Dresden, Germany, September 24 26, 2007.

[8] Resolution A/RES/45/ General assembly of United Nations, Guidelines for the regulation of computerized personal data files, December 1990.

[9] Recommendation of the "Communication of Health Information in Hospitals", European Health Committee CDSP (92)8, Council of Europe, Strasbourg, June 1992.

[10] Recommendations of the Council of Europe, R(97)5, "On The Protection of Medical Data Banks", Council of Europe, Strasbourg, 13 February 1997.

[11] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995, "On the protection of individuals with regard to the processing of personal data", 1995.

[12] European Council, Bangemann report recommendations to the EC, 26 May 1994.

[13] International Risk Governance Council, "Critical infrastructures at risk: Securing the European electric power system", 2007.

[14] North American Electric Reliability Council, "Urgent action standard 1200", 2003.M.A. Harrison, W.L. Ruzzo and J.D. Ullman, "Protection in Operating Systems", Communication of the ACM, 19(8), pp. 461-471, August 1976.

[15] D.F. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn and R. Chandramouli, "A Proposed Standard for RBAC", ACM Tras. on Info. and System Security, v. 4, n 3, August 2001.

[16] A. Abou El Kalam, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, R. El-Baida, A. Mige, C. Saurel, G. Trouessin, "OrBAC: Organization-Based Access Control", 4th International Workshop Policy, Come, Italy, 4-6 June 2003, IEEE Computer Society Press, pp. 120-131.

[17] Law 2002-303 related to the patient's rights and to the quality of healthcare systems, Article L. 1111-7, March 2002.
[18] CISCO, Access Control Lists: Overview and Guidelines, available at http://www.cisco.com/en/US/docs/ios/113/security/configuration/guide/scacls.pdf
[19] L. Aqvist, "Next and Ought, alternative foundations for Von Wright's tense-logic, with an application to deontic logic". Logique & Analyse, vol. 9 (1966) 231–251.
[20] A. Prior, "The paradoxes of derived obligation". Mind vol. 63 (1954) 64–65.
[21] P. Bieber, F. Cuppens, "A definition of secure dependencies using the logic of security", Computer Security Foundations Workshop IV. IEEE (1991).
[22] J. Glasgow, G. MacEwan, and P. Panagaden. "A logic for reasoning about security", ACM Transactions on Computer Science 10, 226264 (1992).
[23] H. Prakken and M. Sergot, "Dyadic deontic logic and contrary-to-duty obligations", In Defeasible Deontic Logic, D.N. Nute (ed), Synthese Library, Kluwer , 223262 (1997)
[24] L. Cholvy and R. Demolombe, "Querying a rule base", first International Conference on Expert Database Systems, Charleston, 1986.
[25] Commission recommendation on collective cross-border management of copyright and related rights for legitimate online music services, 18 May 2005.