Information sharing policies for coalition systems

Laurence Cholvy[1], Christophe Garion[2], and Claire Saurel[1]

[1] ONERA-Toulouse
2bis avenue Édouard Belin
31055 Toulouse Cedex 4
France
{cholvy, saurel}@cert.fr

[2] SUPAERO
10 avenue Édouard Belin
31055 Toulouse Cedex 4
France
garion@supaero.fr

The aim of this paper is to define a logical language to express information sharing policies for coalitions, which have to cope with dynamical environments. We propose to use a first-order logic base language to express policies via concepts like time, action, context, roles in organizations and deontic notions. We define then consistency for a sharing policy and propose two definitions for policy completeness.

1 Introduction

In a coalition, command and control units of different countries need to share information coming from lots of sources (such as intelligence sources or other ones), in order to get for instance a common representation of the crisis situation, and then take relevant decisions to achieve their mission. They also have to cope with amounts of pieces of partial information, with short information processing time limits. Moreover, such information sharing takes place in a high risk environment [14]:

- countries involved in a coalition are not necessary allies,
- trust relation between them may change over the time,
- trust relations may be not symmetric between countries,
- people may change their role in the organization of the coalition, and so change their "need to know".

In such conditions, there is quite a big threat of violating information security properties, such as confidentiality (no unauthorized divulging of secrete information) or availability (information must be available according to users' rights). This may have disastrous consequences for each country's national security.

So, in order for users to trust an information exchange system such as a COP (Common Operational Picture) [1], it is necessary to control and regulate information broadcast within the system.

Given a distributed information exchange system to be designed, one issue is to provide its designers with a *sharing policy* to protect information and, through information, every country involved in the coalition. A sharing policy can be seen as a regulation which specifies authorized, permitted or prohibited diffusion of information within the system.

For example, in such a sharing policy, one could express rules such as:

- *in a context of occurrence of any event related to terrorism in Sweetland, information about this event must be sent to the commander of the joint task force (CJTF) of the coalition K before one hour.* In this rule the context is defined by a kind of event, and information must be sent to someone who plays the role of CJTF in the organization of K.

Cholvy, L.; Garion, C.; Saurel, C. (2006) Information Sharing Policies for Coalition Systems. In *Dynamic Communications Management* (pp. 15-1 – 15-12). Meeting Proceedings RTO-MP-IST-062, Paper 15. Neuilly-sur-Seine, France: RTO. Available from: http://www.rto.nato.int/abstracts.asp.

- *in a context of crisis, any piece of information connected to the topic "aircraft" must, as soon as learned by an agent a, be sent to an agent b within less than 3 seconds.* It is worthwhile sending the piece of information in such a short time, because it will become quickly irrelevant, due to the speed of aircraft.
- *in every context, everybody is forbidden to send pieces of information with a security level of n to anybody whose habilitation level is less than n.* Such a rule may be used in a coalition where each piece of information and each agent are assigned respectively a security and an habilitation level.

A sharing policy can be useful for several issues: specify an information exchange system, increase the trust of its users (denoted agents in the remainder of this paper) and the system reliability, thus making it really useful. For this reason, it is important to get a "good" sharing policy: the quality of a sharing policy depends on some properties such as its consistency or its completeness.

As the subject of this paper is connected with system information security, we take our inspiration from a well known approach in this field, consisting in defining security policies in order to preserve security properties of information (mainly, confidentiality, availability, and integrity). What can we learn from it? On the one hand, since actors of a coalition are often military forces, we could think about using mandatory models [2, 3], where users' rights are defined by their organization. With this approach, rights cannot easily be changed over time and cannot be delegated to other users. On the other hand, discretionary access control models [11] allow each subject (or active entities) to give its access rights on an object (or information container) to other subjects. Unfortunately, they may lead to information leak and so violate confidentiality. Both kinds of models only explicitly regulate permission access to pieces of information, obligation access being implicitly managed through the information system specifications. However the previous examples of rules show that we need obligation rules for information diffusion, at least for information relevance and availability reasons. Moreover, the rules defining obligation about sharing must be explicit, in order to be able to verify some properties on the whole set of sharing rules.

The aim of this paper is first to define a formalism to help one to express a sharing policy (section 2). This formalism will be based upon deontic concepts and first order logic. We define then within this framework the properties of consistency and completeness for a sharing policy in sections 3 and 4. We will then sketch further issues for this preliminary work.

2 A formalism for expressing information sharing policies

In this section, we will present the concepts used in our formalism, a logical framework to represent and to reason about them and a method for expressing policies.

2.1 Useful concepts

In order to express a sharing policy, we need the following primitive concepts: time, actions, properties, deontic modalities and contexts. We will present them in the following.

Time is an important concept, because the deontic notions associated with information sharing will change over time. We need to distinguish three temporal dimensions:

- the time at which an information is valid,

- the time at which an agent gets an information,
- the time at which an agent sends an information.

Those three notions are necessary. For instance, we may express that an agent is obliged to send an information as soon as he *gets* it and to send it before a certain amount of time. In this case, we have to know the time at which the agent sends the information in order to verify that he has not violated the previous obligation.

We will consider only two actions in our framework:

- *learn(x, i, t)* which means that agent x learns information i at time t,
- *send(x, i, y, t)* which means that agent x sends information i to an agent y at time t.

Properties represent general assertions about the current state of the world. We can distinguish the time-dependent properties form the others. For instance:

- *Ally(x, y, t)*: the country of the agent y is an ally of the country of the agent x at time t,
- *Level(x, l, t)*: the habilitation level (respectively the security level) of an agent (respectively an information) x is l at time t. In information security, the definition of *Level* values for military context is often based upon a lattice. For instance, we can distinguish "Classified" and "Top-secret" information and express with the lattice that classified information is more confidential than top-secret information.
- *Topic(i, to)*: the information i deals with topic to.
- *Playsrole(x, r, o, t)*: the agent x plays the role r in the organization o at time t.
- *Hsuperior(x, y, o, t)*: the role x is hierarchically superior to the role y in the organization o at time t.

As we want to express norms, i.e. rules which specify what must, may or must not be done, we need deontic modalities, particularly about information sharing. Therefore, we introduce classical deontic concepts of obligation, prohibition and permission for information sending:

- *Obligatory(send(x, i, y, t))* means that agent x is obliged to send the information i at time t to agent y,
- *Prohibited(send(x, i, y, t))* means that agent x is prohibited to send the information i at time t to agent y,
- *Permitted(send(x, i, y, t))* means that agent x is permitted to send the information i at time t to agent y.

Moreover, we need consistency axioms between these deontic predicates, which are expressed through the following constraints[1]:

- $\neg(Permitted(x) \land Forbidden(x))$
- $Obligatory(x) \rightarrow Permitted(x)$
- $\neg(Obligatory(x) \land Forbidden(x))$

[1] Notice that the third one can be deduced from the first two.

Classically, those constraints express the fact something obligatory or permitted cannot be forbidden, and that something obligatory is permitted.

Finally, notion of context is important here. Coalitions work in dynamical environments: crisis, quiet situations, occurrences of events etc. For instance:

- *Occurrence(e, k, t)* means that an event e of kind k occurs at time t.
- *Crisis(t)* means that there is a crisis situation at time t.

Information sharing modalities may depend upon each kind of environment, and we will call this environment *context*. Contexts will be mentioned in information sharing rules. Therefore, if c is a context, a general form for a rule r is: $c \rightarrow r$ meaning that rule r applies in context c.

2.2 A logic-based formalism

In this section we propose a logical framework L in order to deal with the concepts defined previously. This framework is based upon a typed first order logic.

We suppose that atomic pieces of shared information are expressed through a given entity-relation database model. That kind of representation is currently used in the coalition field, see for instance the IDEF1X (Integration Definition for Information Modeling) language on which is based the JC3IEDM (Joint Control Command and Consultation Information Exchange Data Model) formalism. In such models, entities represent a kind of “type” (like aircraft) and entity instance a particular object of this type. Objects can be composite objects: for instance coordinates are composed of two numbers. Relations is an association between entities. For instance, *position* is a relation between entity *aircraft* and entity *coordinates*.

A strong hypothesis in this work is that we will only deal with atomic informations, like *the position of the object O is* $(45, 32)$. This seems to be sufficient for our application needs.

As usual, the alphabet of L will be based on three distinct groups of symbols: constant symbols, predicate symbols and function symbols.

Let us precise that constant names will be denoted by upper Latin symbols (object O, agent A), whereas variables will be denoted by lower Latin symbols. Moreover, predicate names will begin by an upper symbol and function names by lower symbol.

Finally, as we want to type our language, we will distinguish different groups of symbols among those three categories.

Definition 1. *We distinguish four sets of constants:*

- I-constants *which represent values of the domain of the attributes of the information data-base model.*
- Ag-constants *which represent agents who share information in the system.*
- T-constants *represent time points (essentially as dates).*
- *other constants will be denoted by* O-constants.

In the previous example, object O, 45 and 32 are I-constants. O-constants are used to represent information topic (localization in our example), or security levels for instance.

Definition 2. *We characterize predicate symbols in the following way:*

- *Obligatory, Permitted and Forbidden are unary predicates that we call D-predicates (for deontic predicates).*
- *Learn(.,.,.) is a ternary predicate symbol.*
- *contexts are expressed through predicates with at least one parameter for time. We will note them C-predicates : Crisis (.), Occurrence (.) . . .*
- *P-predicates will be used to express any kind of property on informations, agents, etc.*

We use predicates to represent deontic notions as in [9]. Notice that P-predicates include the classical mathematical operators like $>$ and $=$. Some other examples of P-predicates are : *Playsrole(.), Level(.,.,.), Ally(.,.,.)* etc.

Definition 3. *Functions are characterized in the following way:*

- *I-functions represent relations of the information data base model with corresponding arity.*
- *not(.) is a unary-function used to represent object level negation.*
- *send(.,.,.,.) is a function with four arguments representing the action of sending an information.*

For instance, the position relation in the database is represented by the I-function *position(.,.)*. We can now define formulas for L.

Definition 4. *Formulas of L are defined recursively as follows:*

- *If f is a I-function, if $t_1, \ldots, t_n$ are I-constants or variables, then $f(t_1, \ldots, t_n)$ and $not(f(t_1, \ldots, t_n))$ are I-terms.*
- *If $t_1, \ldots, t_n$ are constants or variables, if C is a C-predicate, then $C(t_{1,\ldots,t_n}$ is a C-literal and is a formula of L.*
- *Let x be an Ag-constant, i be an I-term or a variable, t be a T-constant or a variable. Then $Learn(x, i, t)$ is a L-literal and a formula of L.*
- *Let x and y be Ag-constants or variables, i be an I-term or a variable, t be a T-constant or a variable. Then $Obligatory(send(x, i, y, t))$, $Permitted(send(x, i, y, t))$ and $Forbidden(send(x, i, y, t))$ are D-literals. They are formulas for L.*
- *If $t_1, \ldots, t_n$ are constants or variables, if P is a P-predicate, then $P(t_1, \ldots, t_n)$ is a P-literal, and a formula of L.*
- *Let F_1 and F_2 be formulas of L and x be a variable. Then $\neg F_1$, $F_1 \wedge F_2$, $F_1 \vee F_2$, $\forall x\ F_1$ and $\exists x\ F_1$ are formulas of L, as it is usually defined.*

2.3 Definition of an information sharing policy

In this section, we define rules for an information sharing policy, within the above logical language.

An information sharing policy is a set of formulas of L which are Horn clauses[2] $l_1 \vee l_2 \vee \ldots \vee l_n$ such that:

- l_n is the only positive literal and is a D-literal,
- $\forall i \in \{1, \ldots, n-1\}$, l_i is a negative C-literal, L-literal, P-literal or D-literal,

[2] An Horn clause is a clause in which only a literal is positive.

- if x is a variable in l_n, then $\exists i \in \{1, \ldots, n-1\}$ such that l_i is a negative literal and contains the variable x. This last condition comes from the definition of *restrictive field* in the data bases domain: it aims to characterize significative formulas.

Rules of sharing policy can be be expressed by such formulas.

Example 1. The rule "*in a context of crisis and occurrence of any event related to terrorism in Sweetland, information about this event must be sent to the commander of the joint task force (CJTF) of the coalition K before one hour later*" is expressed with the following formula:

$$\begin{aligned}(R0)\ &\forall i\ \forall t\ \forall t'\ \forall x\ \forall y\ \forall e\ \ Crisis(t) \wedge Playsrole(x, CJTF, K, t)\\ &\wedge occurrence(e, Terrorist, t) \wedge Learn(y, position(e, Sweetland), t') \rightarrow\\ &Obligatory(send(y, position(e, Sweetland), x, t'+1))\end{aligned}$$

Example 2. Suppose our policy deals with confidentiality for informations with a multi level model; suppose that in that model, each piece of information and each agent are assigned respectively a security and an habilitation level.

The rule "*in every context, everybody is forbidden to send pieces of information with a security level of n to anybody whose habilitation level is less than n*" is expressed with the following formula:

$$\begin{aligned}(R1)\ &\forall x\ \forall i\ \forall y\ \forall n\ \forall n'\ \forall t_0\ \forall t_1\ \forall t_2\ \forall t_3\ Learn(x, i, t_0) \wedge Learn(x, level(i, n), t_1) \wedge\\ &Learn(y, level(y, n'), t_2) \wedge (n' < n) \wedge (t_3 > \max(t_0, t_1, t_2)) \rightarrow\\ &Forbidden(send(x, i, y, t_3))\end{aligned}$$

Notice that there is no context predicate in this formula, so the rule is applicable in every context.

Consider now a sharing policy saying that *in a context of crisis, any information about the topic "air-ground missile (AGM)" must, as soon as learned by agent A, be sent to agent B.*

This rule may be expressed with the following formula:

$$\begin{aligned}(R2)\ &\forall i\ \forall t\ \forall t'\ \ Crisis(t) \wedge Learn(A, i, t) \wedge Learn(x, topic(i, AGM), t') \rightarrow\\ &Obligatory(send(A, i, B, max(t, t')))\end{aligned}$$

3 Consistency of an information sharing policy

Given a situation and a sharing policy, we want to avoid to deduce that some agent a is both obligated and prohibited (or permitted and prohibited) to send an information to some other agent b. In such cases, it would be impossible for a to know what it has to do. In other words, a would have to face up with a dilemma. Therefore, we will classically define the property of consistency for a sharing policy.

Let *Dom* be the set of domain knowledge, and domain meta-knowledge. For instance, it includes relations between topics concerned by information. *Dom* may for instance include following knowledge:

$$(D1)\ \forall x\ \forall y\ \forall z\ type(x,y) \rightarrow topic(type(x,y),y) \wedge topic(position((x,z),y))$$

(D1) means that if the type of x is y, then the information "the type of x is y" and "the position of x is z" deal both with the topic y.

$$\begin{aligned}(D2)\ \forall x\ \forall y\ \forall z\ \forall a\ \forall t\ & Learn(a, type(x,y),t) \rightarrow Learn(a, topic(type(x,y),y),t) \wedge \\ & Learn(a, topic(position(x,z),y),t)\end{aligned}$$

(D2) means that if an agent a learns at time t that the type of x is y, then at the same time a learns that the information "the type of x is y" and "the position of x is z" deal both with the semantic topic y.

$$(D3)\ \forall t\ \neg(Quiet(t) \wedge Crisis(t))$$

(D3) means that a context cannot be both quiet and a crisis context.

$$(D4)\ Majorlevel(SD, CD)$$

(D4) means that the *SD* (Top-secret) habilitation or security level is greater than the *CD* (Confidential) habilitation or security level, in the case of a multilevel application.

Let us also add the following axioms about D-predicates, as stipulated in 2.1:

$$\begin{aligned}&(A1)\ \forall x\ \neg(Permitted(x) \wedge Forbidden(x))\\&(A2)\ \forall x\ Obligatory(x) \rightarrow Permitted(x)\\&(A3)\ \forall x\ \neg(Obligatory(x) \wedge Forbidden(x))\end{aligned}$$

$(A1)$ means that nothing cannot be both permitted and forbidden. $(A3)$ means that nothing cannot be both obligatory and prohibited and $(A2)$ means that anything which is obligatory has also to be permitted. Notice that we can deduce $(A3)$ from $(A1)$ and $(A2)$.

We can now introduce our definition of consistency for a policy.

Definition 5. *Let P a sharing policy, defined as a set of formulas of L (cf. 2.2). P is said to be* consistent *if and only if there does not exist any set S of clauses without D-literal such that the logical theory $P \cup \{(A1), (A2), (A3)\} \cup S \cup Dom$ is inconsistent.*

If we are able to find such a set S, then S is the set of circumstances that can lead to a contradiction.

We will next illustrate this definition through two examples.

Example 3. Let P a sharing policy which says that in a crisis context:

– ($R2$) any agent x must send to every agent y every piece of information dealing with the topic AGM as soon as it has learn it:

$$(R2)\ \forall x\ \forall i\ \forall y\ \forall t\ \forall t'\ Crisis(t) \wedge Learn(x, i, t) \wedge Learn(x, topic(i, AGM), t') \rightarrow \\ Obligatory(send(x, i, y, \max(t, t')))$$

– ($R3$) every agent is forbidden to send any information dealing with the topic "Nuclear" (written Nu) to anybody:

$$(R3)\ \forall x\ \forall i\ \forall y\ \forall t\ \forall t'\ \forall t'' Crisis(t) \wedge Learn(x, i, t) \wedge Learn(x, topic(i, Nu), t') \wedge \\ t'' > max(t, t') \wedge \rightarrow Forbidden(send(x, i, y, t''))$$

Let Dom include the rules ($D2$) and ($D3$). Let us now consider the following scenario:

- there is a crisis context.
- on March the 30th, a learns the position of an object o, and learns that o is a nuclear arm.
- on March the 31st, a learns that o is an air-ground missile (AGM).

With P we can deduce that from ($D2$), on March the 30th a learns that the information about the position of o is related to the topic Nu. Thus, from ($R3$), a is forbidden to send the position of o from March the 30th. a is in particular forbidden to send the position of o to the agent b on March the 31st.

However, as a learns on March the 31st that o is an air-ground missile, from ($D2$), a also learns that the piece of information about the position of o is related to the topic AGM. Then from ($R2$), a is immediately obliged to send it to b.

So on March the 31st, the agent a has to face up with a dilemma: to send or not to send the position of o to b.

Let us consider $S = \{Learn(a, type(o, Nu), 30), Learn(a, type(o, AGM), 31), Crisis(30)\}$. We can show that $P \cup \{(A1), (A2), (A3)\} \cup S \cup Dom$ is inconsistent. That means that P is inconsistent according to our previous definition.

Example 4. Let us consider P' composed of two rules:

– ($R2$), about diffusion of AGM information in context of crisis:

$$(R2)\ \forall x\ \forall i\ \forall y\ \forall t\ \forall t'\ Crisis(t) \wedge Learn(x, i, t) \wedge Learn(x, topic(i, AGM), t') \rightarrow \\ Obligatory(send(x, i, y, \max(t, t')))$$

– and ($R4$): in a quiet context, every agent is forbidden to send any information dealing with the semantic topic "Nuclear" (written Nu) to anybody:

$$(R4)\ \forall x\ \forall i\ \forall y\ \forall t\ \forall t'\ Quiet(t) \wedge Learn(x, i, t) \wedge Learn(x, topic(i, Nu), t') \wedge \\ t'' > \max(t, t') \rightarrow Forbidden(send(x, i, y, t''))$$

From ($D3$), we cannot be simultaneously in a quiet and crisis context, there is no situation in which an agent is simultaneously obligated and forbidden to send any information dealing both with AGM and Nu topics. So, according to our definition, P' is consistent.

4 Completeness of an information sharing politics

Now, the intuition we want to capture is that given a sharing policy P, in any situation, P allows to deduce if an agent a is allowed, obligated or forbidden to send a particular information to another agent b.

We propose a first definition:

Definition 6. *Let P be a sharing policy defined on L. P is said to be* complete *if and only if for every context c, every time constant t, every couple of agents a and b, and every information i, the following property is true:*

- $P \models c \rightarrow \mathit{Obligatory}(\mathit{send}(a, i, b, t))$ *or*
- $P \models c \rightarrow \mathit{Forbidden}(\mathit{send}(a, i, b, t))$ *or*
- $P \models c \rightarrow \mathit{Permitted}(\mathit{send}(a, i, b, t))$

In fact, it is quite difficult to anticipate all possible cases while defining a sharing policy: our first definition for completeness is unrealistic. What seems more realistic is to impose completeness only for important subjects or some topics or restrict completeness to a small group of agents. For instance, an agent should always know what to do with an important information. Thus, we propose a weaker definition for completeness.

Definition 7. *Let P be a sharing policy defined on L. Let $D(x, i, y, t)$ a formula of L and C be an information representing a context. P is said to be* complete for D and C *for every couple of agent x and y if and only if:*

- $P \models c \rightarrow (\forall x\ \forall i\ \forall y\ \forall t\ D(x, i, y, t) \rightarrow \mathit{Obligatory}(\mathit{send}(x, i, y, t)))$ *or*
- $P \models c \rightarrow (\forall x\ \forall i\ \forall y\ \forall t\ D(x, i, y, t) \rightarrow \mathit{Forbidden}(\mathit{send}(x, i, y, t)))$ *or*
- $P \models c \rightarrow (\forall x\ \forall i\ \forall y\ \forall t\ D(x, i, y, t) \rightarrow \mathit{Permitted}(\mathit{send}(x, i, y, t)))$

Example 5. Let us resume example 3. The rules for policy are:

$$(R2)\ \forall x\ \forall i\ \forall y\ \forall t\ \forall t'\ \mathit{crisis}(t) \wedge \mathit{Learn}(x, i, t) \wedge \mathit{Learn}(x, \mathit{topic}(i, AGM), t') \rightarrow \\ \mathit{Obligatory}(\mathit{send}(x, i, y, \max(t, t')))$$

$$(R4)\ \forall x\ \forall i\ \forall y\ \forall t\ \forall t'\ \mathit{Quiet}(t) \wedge \mathit{Learn}(x, i, t) \wedge \mathit{Learn}(x, \mathit{topic}(i, Nu), t') \wedge \\ t'' > \max(t, t') \rightarrow \mathit{Forbidden}(\mathit{send}(x, i, y, t''))$$

We can show that this policy is complete for the following formula:

$$\exists t\ \exists t'\ \mathit{Learn}(x, i, t) \wedge \mathit{Learn}(x, \mathit{topic}(i, Nu), t) \wedge t'' > \max(t, t')$$

This means that if an agent a knows an information and learns that this information has Nu as topic, then a knows what to do regarding to sending the information (more precisely, a is forbidden to send the information).

5 Conclusion

In this paper we have defined a logical framework to express and reason about information sharing policies for coalition. The rules expressed in this policy depend on several concepts: deontic notions, such as permission and obligation, time, communication actions and context.

We have proposed a definition for policy consistency. Consistency allows a policy designer to verify that an agent cannot face a dilemma concerning an information. Notice that we can use SOL deduction [12] to verify efficiently this consistency and to find the eventual counterexamples (cf. also [9]).

The completeness problem is different and more difficult. The first definition we proposed is too restrictive: in order to obtain the completeness property, the designer of a policy must know in advance all the possible cases for the policy. We then proposed a restricted definition for completeness allowing to consider the property only for some topics for instance. The designer can concentrate only on the important domains.

This preliminary work can be extended in several directions.

First, we can go deeper on the theoretical framework by proposing a more precise definition for completeness for instance. Notice also that we have not treated the classical problems of deontic logic like Contrary-to-Duties [8, 6]. This study has to be done, particularly in the coalition context where regulation can be huge and where such problems may arise. We can also study obligations with deadline which is strongly related to our problem [5].

The *Learn* predicate semantic must also be studied. More precisely, the formal link between an agent's beliefs base updates (when the agent receives an information) [10] and the norms application (a permission or an obligation has to be taken into account at a certain date) is an interesting extension of this work. If we consider that each agent has a belief base which can be updated by new information, the "triggers" for new regulation has to be calculated from the difference between the agent's old beliefs and new beliefs (only new informations have to be considered).

Finally, in a coalition, the need for information for an agent is more constrained by the agent's role than the agent itself. Several agents can have the same role in the coalition, the role of an agent can change during the coalition mission etc. Thus, we have introduced in our framework the notion of role [4, 7]. Moreover, it can be interesting to use the various works on RBAC (Role-Based Access Control) security policies [15, 13]. Using roles, we can express conditions on the agents' roles, which is less fastidious than expressing conditions on agents (the roles in a coalition are quite stable, whereas the agents can change frequently). Notice also that the notion of role has been used in the architecture for secured information sharing in dynamic coalition presented in [14]. In our formalism, we have introduced roles through the predicates *Playsrole* and *Hsuperior*, but this needs more efforts to have a complete representation of the notions developed in the cited papers.

Acknowledgements

This work was supported by the DGA/SPOTI 03.73.088 research contract.

References

1. Global command and control system common operational picture reporting requirements. `http://www.dtic.mil/cjcs_directives/cdata/unlimit/3151_01.pdf`.

2. D.E. Bell and L.J. LaPadula. Secure computer systems: unified exposition and multics interpretation. Technical report, The MITRE corporation, 1975.
3. K.J. Biba. Integrity consideration for secure computer systems. Technical report, The MITRE corporation, 1977.
4. G. Boella and L. van der Torre. Attributing mental attitudes to roles: The agent metaphor applied to organizational design. In *Proceedings of ICEC'04*. ACM Press, 2004.
5. J. Broersen. On the logic of 'being motivated to achieve ρ before δ'. In J. J. Alferes and J. Leite, editors, *Logics in Artificial Intelligence, 9th European Conference JELIA 2004*, number 3229 in Lecture Notes in Artificial Intelligence, pages 334–346. Springer, 2004.
6. J. Carmo and A. Jones. Deontic logic and contrary-to-duties. In *Handbook of Philosophical Logic*, volume 8: Extensions to Classical Systems 2. Kluwer Publishing Company, 2001.
7. J. Carmo and O. Pacheco. Deontic and action logics for organized collective agency, modeled through institutionalized agents and roles. *Fundamenta Informaticae*, 48(2,3):129–163, 2001.
8. R. Chisholm. Contrary-to-duty imperatives and deontic logic. *Analysis*, 24:33–36, 1963.
9. L. Cholvy. Checking regulation consistency by using SOL-resolution. In *International Conference on Artificial Intelligence and Law*, pages 73–79, 1999.
10. P. Gardenfors. *Knowledge in Flux : Modelling the Dynamics of Epistemic States.* MIT Press, 1988.
11. M.A. Harrison, W.L. Ruzzo, and J.D. Ullman. Protection in operating systems. In *Communications of the ACM*, volume 8, pages 461–471. ACM Press, 1976.
12. K. Inoue. Linear resolution for consequence finding. *Journal of Artificial Intelligence*, 56:301–353, 1992.
13. A. A. El Kalam, R. El Baida, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, A. Miège, C. Saurel, and G. Trouessin. Organisation based access control. In *Proceedings of POLICY 2003, IEEE 4th international workshop on policies for distributed systems and networks*, 2003.
14. C. E. Phillips, T. C. Ting, and S. A. Demurjian. Information sharing and security in dynamic coalitions. In *SACMAT*, pages 87–96, 2002.
15. R. Sandu. Role-based access control. *Advances in computer sciences*, 48, 1998.