Privacy policies with modal logic: the dynamic turn

Guillaume Aucher, Guido Boella, Leendert van Der Torre

Privacy policies with modal logic: the dynamic turn*

Guillaume Aucher[1], Guido Boella[2], Leendert van der Torre[1]

[1]University of Luxembourg. {guillaume.aucher, leon.vandertorre}@uni.lu
[2]Università di Torino - Italy. guido@di.unito.it

Abstract. Privacy policies are often defined in terms of permitted messages. Instead, in this paper we derive dynamically the permitted messages from static privacy policies defined in terms of permitted and obligatory knowledge. With this new approach, we do not have to specify the permissions and prohibitions of all message combinations explicitly. To specify and reason about such privacy policies, we extend a multi-modal logic introduced by Cuppens and Demolombe with update operators modeling the dynamics of both knowledge and privacy policies. We show also how to determine the obligatory messages, how to express epistemic norms, and how to check whether a situation is compliant with respect to a privacy policy. We axiomatize and prove the decidability of our logic.

1 Introduction

Privacy policies are often static and defined as a set of permitted messages, for example in traditional access control languages [2,8,13,19]. If policies were instead defined in terms of the permitted and forbidden knowledge of the resulting epistemic state of the recipient of information, then the permitted messages could be derived by combining and reasoning on this knowledge. This raises the following research problem studied in this paper:

> How to formally specify and reason about privacy policies in terms of permitted and forbidden knowledge?

The challenge in this research problem is that the exchange of messages changes the knowledge, and we therefore need a dynamic language which allows us to reason about these changes. Moreover, we impose the following requirements on languages for specifying and reasoning about privacy policies.

We must be able to distinguish between a permission to know and the permission to send a message. For example, you may be permitted to know your medical file, while it may not be permitted that someone not being a doctor sends you your medical file. How do such distinctions allow for a more fined-grained account of classical problems of security such as the Chinese wall problem?

We must be able to specify and reason about the *order* in which messages can be sent. For example, it may be permitted to send some sensitive information only if a message has been sent before detailing how to deal with sensitive messages. In many

* We thank the anonymous reviewers of this paper for helpful comments.

cases it is more efficient or natural to specify that a given piece of information may not be known, than explicitly forbidding the different ways of communicating it.

We must be able to specify *obligations* in privacy policies. For example, it might happen that some additional instructions *should* be sent to the user about the nature of the previous information he received, or that in case personal information is disclosed inappropriately, the subject of information *should* be informed. As [6] notices, privacy laws actually specify which counter measures should apply in case a situation is not compliant with a privacy policy.

We must be able to express *meta-security policies*. These are regulations about how to access the regulation itself. For instance, in some applications there is a need for constraints of the form: "agents who play the role r_1 are forbidden to know that agents who play the role r_2 are permitted to know p"; these constraints may be fulfilled using "cover stories" to hide some data [15].

We use modal logic, since both knowledge and obligations (and permissions) are traditionally and naturally modeled in branches of modal logic called epistemic and deontic logic respectively. This is no new observation in the area of security: Cuppens already introduced in 1993 a modal logic for a logical formalization of secrecy [14], and together with Demolombe he developed a logic for reasoning about confidentiality [16] and a modal logical framework for security policies [17]. This epistemic deontic logic is the basis of the formalism we introduce in this paper.

The Cuppens-Demolombe logic already got many things right: it can express epistemic norms, i.e. norms regulating what is permitted to know, and can represent violations, unlike most privacy regulation languages (like [6]). However, despite its strengths, it is not able to reason about the dynamics of knowledge and privacy policies, and it does not satisfy the above four requirements. They were ahead of their times, since in 1993 dynamics in modal logic was mainly restricted to propositional dynamic logic for reasoning about programs. In fact the dynamics of knowledge was studied mainly in the AGM paradigm of theory revision [1]. In the meantime, much has changed. Dynamic epistemic logic has become a standard branch of modal logic, on which textbooks have been written [27], and which is taught at many universities. Our modal logic extends the Cuppens-Demolombe logic with dynamic update operators, to model both the dynamics of knowledge and of privacy policies. As Cuppens and Demolombe, we define privacy policies in terms of knowledge that the recipient of information is permitted/prohibited to have. The way we define the dynamics of knowledge then allows us to derive the policies on messages. With this new dynamic feature, we can not only determine in a generic way the permitted sequence of messages in a given situation but also determine which change is needed in order to enforce a (possibly new) privacy policy.

The paper is organized as follows. In Section 2, we describe the range of phenomena under study, and we give a number of examples to provide some intuitions. In Section 3, we introduce our Dynamic Epistemic Deontic Logic (DEDL). We start with the static part, defining epistemic norms and privacy policies, and we then add dynamics, defining permitted (and obligatory) messages and enforcements of privacy policies. Finally, in Section 5, we mention some related works and end with some concluding remarks.

2 Our Scenario of Privacy Regulations

In this paper, we consider a single agent (*sender*) communicating information from a knowledge base to another agent (*recipient*), with the effect that the *recipient* knows the information. The *sender* is subject to privacy regulations which restrict the messages he can send to the *recipient*. We illustrate the distinction between norms of transmission of information and epistemic norms with an example:

Example 1. Consider a *sender* s, e.g., a web server, which is subject to a privacy regulation: he should not communicate the address a of a person to the *recipient* r: we could write this as a norm of transmission of information, regulating the sending of a message: $\neg P_s(send\ a)$, which denotes the permission that the *sender* sends message a. Instead, in an epistemic norm perspective, this prohibition can be derived from the prohibition for the *sender* that the *recipient* comes to know the address: $K_r a$. This is expressed by a deontic operator indexed by the *sender* and having as content the ideal knowledge K'_r of the *recipient*: $\neg P_s K'_r a$.

This distinction is bridged by modelling sending actions performed by the *sender* which update the knowledge of the *recipient*.

Example 2. The message sending action $[send\ a]$ expresses that the *sender* sends to the *recipient* the address a. The result of the action is that the *recipient* knows a: $K_r a$. Since $K_r a$ is not permitted by the epistemic norm $\neg P_s K'_r a$, the *sender* during his decision process derives that also the action $[send\ a]$ is not permitted: $\neg P_s(send\ a)$. Analogously, all other possible actions leading to the forbidden epistemic state $K_r a$, if any, are prohibited too. E.g., if the address is composed by street e, number n and town t: $e \wedge n \wedge t \leftrightarrow a$, then the sequence of messages $[send\ e][send\ n][send\ t]$ leads to the forbidden epistemic state $K_r a$.

While we need to explicitly model the knowledge of the *recipient* resulting from the message, it is not necessary to have an explicit modality for the *sender*, since we have only one *sender* and we adopt his point of view. So a alone means that the *sender* knows the address. This explains also why we talk about "knowledge" of the *recipient*: the *sender* never lies, so the result of his actions on the epistemic state of the *recipient* is knowledge rather than belief: $K_r a$ implies a, i.e., that the *sender* holds a as true. If instead we allowed the *sender* to lie to protect some secrets (as, e.g., [10] do), then the result of the action of sending messages would be a mere belief of the *recipient*: the result of $[send\ a]$ would be that the *recipient* believes a, but a - from the point of view of the *sender* - would not follow from this.

A logical approach to privacy provides a natural solution to the so-called inference problem, i.e. how further permissions propagate from permitted information:

Example 3. Assume it is prohibited to know the street where some person lives. Thus, it must be prohibited to know the address of this person. If $e \wedge n \wedge t \leftrightarrow a$, then $\neg P_s K'_r e$ implies $\neg P_s K'_r a$. Viceversa, if it is permitted to know the address, then it must be permitted to know the street. The same kind of reasoning is transferred at the level of norms of transmission of information: e.g., $\neg P_s(send\ e)$ implies $\neg P_s(send\ a)$, if it is prohibited to send the name of the street, it is prohibited to send the entire address.

Note that to attribute knowledge to the *recipient*, it is neither necessary to have user profiles nor to have any uncertainty. This stems from the assumption that the *sender* is the only source of information for the *recipient* from the knowledge base. The only knowledge that should be considered is the one derived from the past interaction between the two agents, i.e., the information already disclosed by the *sender*. Assuming for simplicity that the *sender* is rational and sends only information consistent with his previous communicative acts, there is no need of some kind of belief revision.

When the forbidden state is achieved by a sequence of messages, there is the possibility that each message of the sequence is permitted while the resulting state is prohibited: this is a new kind of the Chinese wall problem.

Example 4. (Website example) Consider the information about websites contacted by a user (the *recipient*), which are available on a server (the *sender*) logfile. The list of websites for each user is clearly a sensitive information which he would not like to disclose. However, knowing which websites have been visited is a valuable information, for example, for the configuration of a firewall, or to make statistics. Thus it has become anonym by replacing the names of the users with numbers by means of a hashcode (h). So even if one knows the list of users one cannot understand who contacted which website. However, from the association between users and numbers and between numbers and websites the original information can be reconstructed. Therefore the mappings from the users to the numbers (c) and from the numbers to the websites (e) can be distributed individually but not altogether since their association would allow to reconstruct the mapping from the users to the websites they visited (v): $c \wedge e \rightarrow v$.

A solution to enforce this privacy policy could be to forbid the distribution of a mapping if the other one has been already distributed, using a language like the one proposed by Barth *et al.* [6], which is able to express policies about the flow of information referring to actions already performed. This solution, however, requires two rules corresponding to the possible permutations of communicative acts. Moreover, this solution is not general, because there can be further ways of making the forbidden information available. E.g., by distributing the hash function h used. Expressing a flexible policy on all the alternative combinations of actions becomes soon unfeasible. Moreover, new ways of computing the forbidden information could be devised later, which would not be taken into account by the policy.

In this situation we have that it is permitted to know the individual pieces of information, but not what is implied by the conjunction of them:

$$P_s K'_r c, P_s K'_r e, \neg P_s K'_r v.$$

It states that it is permitted to know the mapping between users and numbers ($P_s K'_r c$), it is permitted to know the mapping between numbers and websites visited ($P_s K'_r e$) but it is not permitted to know the mapping between users and their websites visited ($\neg P_s K'_r v$). We have the same situation from the point of view of permissions concerning actions: it is permitted to send the messages c and e individually, but not their combination: $P_s(send\ c) \wedge P_s(send\ e)$ but $\neg P_s(send\ (e \wedge c))$ otherwise the epistemic norm $\neg P_s K'_r v$ would be violated. This means that after sending one of the two messages, the other one becomes prohibited: $[send\ e] \neg P_s(send\ c)$ and $[send\ c]\ \neg P_s(send\ e)$.

The possibility of nesting formulas with epistemic and deontic modalities allows us to express meta-security, i.e., policies concerning the disclosure of policies, as proposed, e.g., by [10]:

Example 5. Sometimes, informing the *recipient* about the prohibition to send some information might lead him to infer something he should not know. For example, if the *recipient* asks whether a person is a secret agent (p), replying "I cannot tell this to you" to the question makes the *recipient* infer that the person is actually a secret agent, otherwise the answer would have been "no". To avoid this case, it should be prohibited to let the *recipient* know the policy that knowing p is prohibited:

$$\neg P_s K'_r \neg P_s K'_r p$$

In contrast, if a policy is permitted to be known, it can even be communicated to the *recipient*: if $P_s K'_r P_s K'_r p$ then it is permitted to send the message $P_s K'_r p$: $P_s(send\, P_s K'_r p)$. This illustrates also that policies can be the content of messages.

3 Dynamic epistemic deontic logic

The logic for privacy regulation should reason about obligations, permissions, knowledge, and information exchange. To deal with these notions altogether, we first extend in Section 3.1 the logic of Cuppens and Demolombe [17] to a more expressive and flexible logic. This logic is actually based on the well-known deontic logic of Castañeda (see the appendix for more details). In Section 3.2, we then add dynamics to the picture. This allows us to have a more fine-grained account of privacy regulations and to solve the research problems that we mentioned in the introduction.

3.1 'Static' privacy policies

Epistemic Deontic Logic (EDL). Starting from a linguistic analysis, the insight of Castañeda's well known approach to deontic logic is to acknowledge the grammatical duality of expressions depending whether they are within or without the scope of deontic operators [11]. We follow this approach and therefore split our language into two kinds of formulas: circumstances and epistemic practitions. The former cannot be in the scope of an obligation operator O whereas the latter are always within the scope of a deontic operator O. This yields the following language $\mathcal{L}_{EDL} = \mathcal{L}^{\phi}_{EDL} \cup \mathcal{L}^{\alpha}_{EDL}$ whose formulas are denoted ϕ^* in general.

$$\mathcal{L}^{\phi}_{EDL} : \phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid K_r\phi \mid O_s\alpha$$
$$\mathcal{L}^{\alpha}_{EDL} : \alpha ::= K'_r\phi \mid \neg\alpha \mid \alpha \wedge \alpha$$

where p ranges over Φ^{ϕ}. Formulas of $\mathcal{L}^{\phi}_{EDL}$ are called circumstances and formulas of $\mathcal{L}^{\alpha}_{EDL}$ are called epistemic practitions. $O_s\alpha$ reads 'it is obligatory for the *sender* that α'. $P_s\alpha$ is an abbreviation for $\neg O\neg\alpha$ and reads 'it is permitted for the *sender* that α'. Pure circumstances are circumstances without obligation operators $O_s\alpha$. $K_r\phi$ reads 'the *recipient* knows that ϕ'. $K'_r\phi$ also reads 'the *recipient* knows ϕ' but this

knowledge operator is always within the scope of a deontic operator and deals with the ideal knowledge of the *recipient*. This language is strictly more expressive than the language of Cuppens and Demolombe [17], even if the semantics is slightly different.

Definition 1 (Semantics). *An EDL-*model *M is a tuple $M = (W, D, R_r, R'_r, V)$, where W is a non-empty set of possible worlds, $R_r : W \to 2^W$, $R'_r : W \to 2^W$ and $D : W \to 2^W$ are accessibility relations on W, D being serial and R_u, R'_u being reflexive.*[1] *$V : \Phi^\phi \to 2^W$ is a valuation. The truth conditions are defined as follows.*

$$\begin{array}{lll} M, w \models p & \textit{iff} & w \in V(p) \\ M, w \models \phi^* \wedge \psi^* & \textit{iff} & M, w \models \phi^* \textit{ and } M, w \models \psi^* \\ M, w \models \neg\phi^* & \textit{iff} & \textit{not } M, w \models \phi^* \\ M, w \models O_s\alpha & \textit{iff} & \textit{for all } v \in D(w),\ M, v \models \alpha. \\ M, w \models K_r\phi & \textit{iff} & \textit{for all } v \in R_r(w),\ M, v \models \phi \\ M, w \models K'_r\phi & \textit{iff} & \textit{for all } v \in R'_r(w),\ M, v \models \phi \end{array}$$

$M \models \phi$ iff for all $w \in W$, $M, w \models \phi$. (M, w) is called a pointed EDL-model. If $\mathcal{P}$ is a set of formulas, we write $M, w \models c(\mathcal{P})$ iff $M, w \models \phi$ for all $\phi \in \mathcal{P}$.

Obviously, one can map epistemic practitions to circumstances. This mapping $t : \mathcal{L}^\alpha_{EDL} \to \mathcal{L}^\phi_{EDL}$ replaces in an epistemic practition α the ideal knowledge operators K'_r by actual knowledge operators K_r. It is needed in order to check whether obligations are fulfilled: for example $O_s\alpha \wedge \neg t(\alpha)$ means that we are in a violation state. Formally, the mapping function $t : \mathcal{L}^\alpha_{EDL} \to \mathcal{L}^\phi_{EDL}$ is defined inductively as follows:

$$\begin{array}{lcl} t(\neg\alpha) & = & \neg t(\alpha) \\ t(\alpha \wedge \alpha') & = & t(\alpha) \wedge t(\alpha') \\ t(K'_r\phi) & = & K_r\phi. \end{array}$$

Theorem 1 (Soundness, completeness and decidability). *The semantics of $\mathcal{L}_{EDL}$ is* sound and complete *with respect to the logic L_{EDL} axiomatized as follows:*

$$\begin{array}{ll} \mathsf{A}_1 & \textit{All propositional tautologies based on } \Phi^\phi \\ \mathsf{A}_2 & \vdash O_s\alpha \to P_s\alpha \\ \mathsf{A}_3 & \vdash K\phi \to \phi \\ \mathsf{A}_4 & \vdash O_s(\alpha \to \alpha') \to (O_s\alpha \to O_s\alpha') \\ \mathsf{A}_5 & \vdash K(\phi \to \psi) \to (K\phi \to K\psi) \\ \mathsf{R}_1 & \textit{If } \vdash \alpha \textit{ then } \vdash O_s\alpha \\ \mathsf{R}_2 & \textit{If } \vdash \phi \textit{ then } \vdash K\phi \\ \mathsf{R}_3 & \textit{If } \vdash \phi^* \to \psi^* \textit{ and } \vdash \phi^* \textit{ then } \vdash \psi^* \end{array}$$

where K stands for K_r or K'_r. L_{EDL} is also decidable.

Proof. It follows straightforwardly from the Sahlqvist correspondence theorem [9] because Axioms A_2 and A_3 are Sahlqvist formulas. To prove decidability, one can show that L_{EDL} has the finite model property by adapting the selection method [9].

[1] An accessibility relation R is reflexive if and only if for all worlds w, $w \in R(w)$. An accessibility relation R is serial if $R(w) \neq \emptyset$ for all worlds w. See [9] for details.

Privacy policies and compliance in EDL. As discussed by Barth *et al.* [6] in the theory of contextual integrity, privacy norms are relevant only in some context, usually defined by roles played by *sender* and *recipient*. This leads us to define the following notions.

Definition 2 (Epistemic norm and privacy policy). *An* epistemic norm *is a formula of the form $\phi \to O_s\alpha$ or $\phi' \to P_s\alpha'$ where ϕ, ϕ' are pure circumstances and α, α' are epistemic practitions. A* privacy policy *$\mathcal{P}$ is a consistent set of epistemic norms. We abusively write $\phi \in \mathcal{P}$ if there is $\phi \to O_s\alpha \in \mathcal{P}$, and in that case the corresponding α is written α_ϕ.*

Note that permissions concern the knowledge of the *recipient*. This fact should not let the reader think that a privacy policy concerns the behavior of the *recipient*. Indeed, the beliefs of the *recipient* are only modified by actions of the *sender*, so these policies regulate the behavior of the *sender* who might disclose information or not to the *recipient* depending on wether or not this disclosure is in conflict with the privacy policy.

Privacy policies are imposed to the decision maker (*sender*) from a hierarchical superior or set up by himself. They should be enforced in any case. However, this set of epistemic norms is not necessarily complete. As a result, the *sender* can perfectly add other epistemic norms as long as they are consistent with the privacy policy, depending on the particular situation at stake. This leads us to define the following notions of open and closed privacy policies. Intuitively, an open privacy policy is a policy where only the permissions of the security policies hold, everything else being forbidden. A closed privacy policy is a policy where only the prohibitions of the security policy hold, everything else being permitted. These definitions are similar with the definitions of permissive and restrictive approach of Cuppens and Demolombe [17].

Definition 3 (Open and closed privacy policy). *Let $\mathcal{P}$ be a privacy policy.*

- *The privacy policy $\mathcal{P}$ is* open *if for all EDL-model (M, w), if $\mathcal{E}(M, w) \cup \mathcal{P} \nvdash P_s\alpha$, then $M, w \models \neg P_s\alpha$.*
- *The privacy policy $\mathcal{P}$ is* closed *if for all EDL-model (M, w), if $\mathcal{E}(M, w) \cup \mathcal{P} \nvdash \neg P_s\alpha$, then $M, w \models P_s\alpha$.*

$\mathcal{E}(M, w) = \{\phi \in \mathcal{L}_{EL}^{\phi} \mid M, w \models \phi\}$ represents the epistemic state of the recipient.

Note that specifying whether a privacy policy $\mathcal{P}$ is closed or open specifies completely what is permitted and forbidden to know for the *recipient* in the pointed EDL-model (M, w). However, in the general case, the privacy policy $\mathcal{P}$ does not specify all the obligations that should hold in a situation (M, w). This leads us to define two notions of compliance. The first notion of compliance, simply called compliance, just checks wether the obligations $O_s\alpha_\phi$ strictly following from the privacy policy $\mathcal{P}$ given the epistemic state $\mathcal{E}(M, w)$ are fulfilled. The second notion of compliance, called strong compliance, checks whether *all* the obligations are fulfilled.

Definition 4 (Compliance). *Let (M, w) be a pointed EDL-model and $\mathcal{P}$ a privacy policy.*

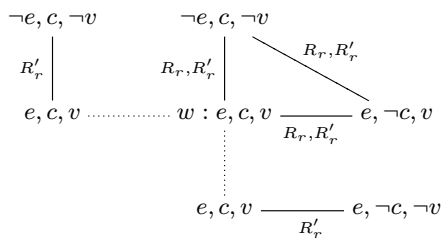

Fig. 1. Website example

- *The situation* (M, w) *is* compliant *with respect to* $\mathcal{P}$ *if* $M, w \models c(\mathcal{P})$ *and* $M, w \models \phi \rightarrow t(\alpha_\phi)$ *for all* $\phi \in \mathcal{P}$.
- *The situation* (M, w) *is* strongly compliant *with respect to* $\mathcal{P}$ *if* $M, w \models c(\mathcal{P})$ *and* $M, w \models O_s\alpha \rightarrow t(\alpha)$ *for all* $\alpha \in \mathcal{L}^{\alpha}_{EDL}$.

The following proposition shows that the distinction between compliance and strong compliance is not relevant for closed privacy policies. It also gives a semantic counterpart to the syntactic notion of strong compliance: an epistemic state (represented by $R_r(w)$) is strongly compliant if there exists a corresponding ideal epistemic state (represented by $R'_r(v)$ for some $v \in D(w)$) containing the same information (i.e. R_rD-bisimilar).

Proposition 1. *Let* (M, w) *be a pointed* EDL*-model and* $\mathcal{P}$ *a privacy policy.*

- *If* $\mathcal{P}$ *is closed then* (M, w) *is compliant w.r.t.* $\mathcal{P}$ *if and only if* (M, w) *is strongly compliant w.r.t.* $\mathcal{P}$.
- *The situation* (M, w) *is strongly compliant w.r.t.* $\mathcal{P}$ *if and only if there exists* $v \in D(w)$ *such that* $R_r(w)$ *and* $R'_r(v)$ *are* R_rD*-bisimilar*[2].

Example 6. (Website example continued) Consider Example 4, where we have the mappings from the users to the numbers (c) and from the numbers to the websites (e), the related mapping from the users to the websites they visited (v) such that $c \wedge e \rightarrow v$. The epistemic norm solution is to express the *privacy policy* $\mathcal{P}_1$ as:

$$\mathcal{P}_1 = \{P_sK'_rc, P_sK'_re, \neg P_sK'_rv\}$$

The pointed EDL-model (M, w) of Figure 1 represents semantically a situation which is *compliant* with respect to this privacy policy. The accessibility relations R_r and R'_r are indexed by R_r and R'_r respectively and the accessibility relation D is represented by dashed arrows. Reflexive arrows are omitted, which means that for all worlds $v \in M$ we also have that $v \in R_r(v)$, $v \in R'_r(v)$ and $v \in D(v)$. We also have that $M \models c \wedge e \rightarrow v$.

[2] Two pointed models (M, v) and (M', v') are R_rD-bisimilar if there is a relation on $W \times W'$ satisfying the base condition for Φ^ϕ and the back and forth conditions for R_r and D (see Blackburn *et al.* [9] for details). If S is a set of worlds of M and S' a set of worlds of M', S and S' are R_rD-bisimilar if and only if for all $v \in S$ there is $v' \in S'$ such that (M, v) is bisimilar to (M', v'), and vice versa.

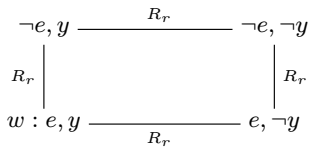

Fig. 2. Spyware example

Example 7. (Spyware example) Consider a situation where the list of websites mentioned is e and the fact that websites might contain risky softwares is y. The privacy policy is expressed by a unique epistemic norm:

$$\mathcal{P}_2 = \{y \wedge K_r e \rightarrow O_s K'_r y\}$$

It states that if the *recipient* knows a list of websites ($K_r e$) which might contain some risky softwares (y), then the *recipient* should know that some of these websites might contain some risky softwares ($O_s K'_r y$). Note that the condition of this epistemic norm contains an epistemic formula. In Figure 2 is depicted a situation compliant with this privacy policy. In this pointed EDL-model (M, w), the accessibility relation R_r is indexed by R_r and reflexive arrows are omitted, which entails that for all $v \in M$, we have $v \in R_r(v)$ and $\{v\} = R'_r(v)$, $\{v\} = D(v)$. We do have that the situation is compliant with respect to the privacy policy $\mathcal{P}_2$.

In fact, we can generalize this kind of policies to stronger policies where the *sender* has to inform the *recipient whether* some information has some property or not.

3.2 The dynamic turn

Dynamic Epistemic Deontic Logic (DEDL). We now want to add dynamics to the picture by means of messages sent to the *recipient*. The content of these messages can affect the situation in two ways: either it affects the epistemic realm (represented in a EDL-model by the relation R_r) or it affects the normative realm (represented in a EDL-model by the relations R'_r and D). This leads us to enrich the language $\mathcal{L}_{EDL}$ with two dynamic operators $[send\ \phi]$ and $[prom\ \alpha]$, yielding the language $\mathcal{L}_{DEDL}$, whose formulas are denoted ϕ^*:

$$\mathcal{L}^{\phi}_{DEDL} : \phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid K_r\phi \mid O_s\alpha \mid [send\ \phi]\phi \mid [prom\ \alpha]\phi$$
$$\mathcal{L}^{\alpha}_{DEDL} : \alpha ::= K'_r\phi \mid \neg\alpha \mid \alpha \wedge \alpha \mid [send\ \phi]\alpha \mid [prom\ \alpha]\alpha$$

where p ranges over Φ^{ϕ}.$[send\ \psi]\phi$ reads 'after the *recipient* learns ψ, ϕ holds', and $[prom\ \alpha]\phi$ reads 'after the *sender* promulgates α, ϕ holds'. The semantics of these dynamic operators is inspired by Kooi [20] and defined as follows.

Intuitively, after learning ψ, the *recipient* restricts his attention to the worlds accessible from the current world which satisfy ψ, unless ψ is not true in this current world. In that case, the message is just ignored. But this second case actually never occurs

here because we assume that *sender* only sends truthful messages. Likewise, after the promulgation of α, the ideal worlds are restricted to the worlds which satisfy α, unless the imperative α is not permitted.

Definition 5 (Semantics). *Let $M = (W, D, R_r, R'_r, V)$ be an EDL-model, $\psi \in \mathcal{L}^{\phi}_{EDL}$ and $\alpha \in \mathcal{L}^{\alpha}_{EDL}$. We define the EDL-models $M * \psi$ and $M * \alpha$ as follows.*

- $M * \psi = (W, D, R^*_r, R'_r, V)$ *where for all* $w \in W$,
  $R^*_r(w) = \begin{cases} R_r(w) \cap ||\psi|| & \text{if } M, w \models \psi \\ R_r(w) & \text{otherwise.} \end{cases}$
- $M * \alpha = (W, D^*, R_r, R'_r, V)$ *where for all* $w \in W$,
  $D^*(w) = \begin{cases} D(w) \cap ||\alpha|| & \text{if } M, w \models P_s\alpha \\ D(w) & \text{otherwise.} \end{cases}$

where $||\phi^*|| = \{v \in M \mid M, v \models \phi^*\}$. The truth conditions are defined as follows.

$$\begin{array}{lll} M, w \models [send\ \psi]\phi^* & \textit{iff} & M * \psi, w \models \phi^* \\ M, w \models [prom\ \alpha]\phi^* & \textit{iff} & M * \alpha, w \models \phi^*. \end{array}$$

Theorem 2 (Soundness, completeness and decidability). *The semantics of $\mathcal{L}_{DEDL}$ is* sound and complete *with respect to the logic L_{DEDL} axiomatized as follows:*

L_{EDL}	*All the axiom schemes and inference rules of* L_{EDL}
A_6	$\vdash [send\ \psi]K_r\phi \leftrightarrow (\psi \rightarrow K_r\,(\psi \rightarrow [send\ \psi]\phi)) \wedge (\neg\psi \rightarrow K_r[send\ \psi]\phi)$
A_7	$\vdash [send\ \psi]K'_r\phi \leftrightarrow K'_r[send\ \psi]\phi$
A_8	$\vdash [send\ \psi]O_s\alpha \leftrightarrow O_s[send\ \psi]\alpha$
A_9	$\vdash [prom\ \alpha]K_r\phi \leftrightarrow K_r[prom\ \alpha]\phi$
A_{10}	$\vdash [prom\ \alpha]K'_r\phi \leftrightarrow K'_r[prom\ \alpha]\phi$
A_{11}	$\vdash [prom\ \alpha]O_s\alpha' \leftrightarrow (P_s\alpha \rightarrow O_s(\alpha \rightarrow [prom\ \alpha]\alpha'))$ $\wedge\,(\neg P_s\alpha \rightarrow O_s[prom\ \alpha]\alpha')$
A_{12}	$\vdash \Box p \leftrightarrow p$
A_{13}	$\vdash \Box\neg\phi^* \leftrightarrow \neg\Box\phi^*$
A_{14}	$\vdash \Box(\phi^* \rightarrow \psi^*) \rightarrow (\Box\phi^* \rightarrow \Box\psi^*)$
R_4	*If* $\vdash \phi^*$ *then* $\vdash \Box\phi^*$

where $\Box$ stands for $[send\ \psi]$ or $[prom\ \psi]$. L_{DEDL} is also decidable.

Proof. We first prove a lemma.

Lemma 1. *For all $\phi \in \mathcal{L}^{\phi}_{DEDL}$ there is $\phi' \in \mathcal{L}^{\phi}_{EDL}$ such that $\vdash \phi \leftrightarrow \phi'$. For all $\alpha \in \mathcal{L}^{\alpha}_{DEDL}$ there is $\alpha' \in \mathcal{L}^{\alpha}_{EDL}$ such that $\vdash \alpha \leftrightarrow \alpha'$.*

Proof (Lemma). First, note that if ψ is a formula without dynamic operator then one shows by induction on ψ using A_6 to A_{14} that $\Box\psi$ is provably equivalent to a formula ψ' without dynamic operator. Now if ϕ is an arbitrary formula with n dynamic operators, it has a subformula of the form $\Box\psi$ where ψ is without dynamic operators which is equivalent to a formula ψ' without dynamic operators. So we just substitute $\Box\psi$ by ψ' in ϕ and we get a provably equivalent formula thanks to A_{14} and R_4 with $n-1$ dynamic operators. We then iterate the process.

As usual in dynamic epistemic logic, we use the previous key lemma to prove the theorem. The soundness part is routine. Let $\phi \in \mathcal{L}_{DEDL}$ such that $\vdash \phi$. Then there is $\phi' \in \mathcal{L}_{EDL}$ such that $\vdash \phi \leftrightarrow \phi'$ by Lemma 1, and therefore $\models \phi \leftrightarrow \phi'$ by soundness. But $\models \phi'$ by Theorem 1, so $\models \phi$ as well. Decidability is proved similarly.

For example, we have the following theorem: $\vdash \psi \rightarrow [send\ \psi]K_r\psi$ for all propositional formula ψ, i.e. after the *sender* sends any truthful message to the *recipient*, the *recipient* knows this message.

Permitted and obligatory messages. Obviously, given a privacy policy and a situation, some messages might not be permitted by the privacy policy because they might lead to a non-compliant situation.

Definition 6 (Permitted and obligatory message). *Let $\phi \in \mathcal{L}^{\phi}_{DEDL}$, $\mathcal{P}$ be a privacy policy and (M, w) an EDL-model representing a given situation.*

- *It is* permitted *for the sender to send message ϕ according to $\mathcal{P}$ in (M, w), written $M, w \models P_s(send\ \phi)$, if $(M * \phi, w)$ is compliant with respect to $\mathcal{P}$.*
- *It is* obligatory *for the sender to send message ϕ according to $\mathcal{P}$ in (M, w), written $M, w \models O_s(send\ \phi)$, if $M, w \models O_sK'_r\phi \wedge \neg K_r\phi \wedge P_s(send\ \phi)$.*

Note also that if it is obligatory to send a message in a situation then this situation is not *strongly* compliant.

Example 8. (Website example continued) In Example 6, we have:

$$M, w \models P_s(send\ c) \wedge P_s(send\ e).$$

So it is permitted to send the mappings from the users to the numbers (c) and it is permitted to send the mapping from the numbers to the web-sites (e). However, we also have

$$M, w \models [send\ e]\neg P_s(send\ c) \text{ and } M, w \models [send\ c]\neg P_s(send\ e)$$

which means that after sending the mapping from the numbers to the web-sites (e) it is *not* permitted to send the mapping from the users to the numbers (c), and vice versa for the second conjunct. This is because in both cases we would violate the epistemic norm $\neg P_sK'_rv$:

$$M, w \models [send\ e][send\ c](K_rv \wedge \neg P_sK'_rv) \text{ and}$$

$$M, w \models [send\ c][send\ e](K_rv \wedge \neg P_sK'_rv).$$

We also have

$$M, w \models \neg P_s(send\ (e \wedge c)).$$

Our approach is very flexible because it is applicable in infinitely many other contexts than the one of the above example, once the privacy policy is fixed. For example, assume that the hash function computing the mapping from users to numbers is now

$$w : e, y \xrightarrow[R_r]{} e, \neg y$$

Fig. 3. Spyware example updated

available (h) and that the *recipient* is able to apply it to get the mapping from numbers to users (c):

$$M \models h \rightarrow c.$$

Applying the same reasoning, we would get:

$$M, w \models [send\ e]\neg P_s(send\ h)$$

$$M, w \models \neg P_s(send\ (e \wedge h))$$

and so without having to introduce explicitly new prohibitions or permissions on h.

Privacy policies do not only concern which information can be disclosed but also which information *should* be disclosed. We can express such policies due to the fact that our epistemic deontic logic can express obligations about knowledge, unlike the one of Cuppens and Demolombe:

Example 9. (Spyware Example continued) After sending the message e in the previous situation represented by the pointed EDL-model (M, w) of Figure 2 we obtain the pointed EDL-model $(M*e, w)$ depicted in Figure 3. The corresponding situation $(M*e, w)$ is not compliant with respect to $\mathcal{P}'$. Therefore, it was forbidden to disclose e:

$$M, w \models \neg P_s(send\ e)$$

But it is now obligatory (with respect to $\mathcal{P}'$) to disclose y:

$$M * e, w \models O_s(send\ y)$$

So we have that

$$M, w \models [send\ e]O_s(send\ y)$$

$$M, w \models \neg P_s(send\ e) \wedge P_s(send\ (e \wedge y)).$$

As it turns out, after sending the message y we reach a compliant situation.

The above example suggests that even if it is prohibited to send message e, it might still be permitted to send message e as long as it is followed by another message y. We leave the investigation of the permissibility of iterative messages for future work.

In privacy regulations, the permission to disclose the names of users also allows to disclose their family names (which are part of their name). This problem, discussed in Example 3, is known as the inference problem, and is in general difficult to model (see for instance Barth *et al.* [6]). In our logical framework it follows easily from the fact that the *recipient* has reasoning capabilities. Indeed, if we assume that the conditions of the epistemic norms of the privacy policy $\mathcal{P}$ are propositional then for all $\phi, \phi' \in \mathcal{L}^{\phi}_{DEDL}$,

$$\phi \to \phi' \models^g P_s(send\ \phi) \to P_s(send\ \phi')$$

where $\models^g$ is the global consequence relation (see [9] for details).

Example 10. (Website example continued) Assume we have a situation modeled by an EDL-model M such that $M \models v \to v'$: the association between the users' name and the web-sites they visited (v) induces the association between the users' *family* name and the web-sites they visited (v'). So if $M, w \models P_s(send\ v)$ then $M, w \models P_s(send\ v')$: if it is permitted to disclose the name of the users in association with the websites they visited, it is also permitted to disclose their family name in association with the websites they visited. Dually, if $M \models v \to v'$, then $M, w \models \neg P_s(send\ v')$ implies $M, w \models \neg P_s(send\ v)$: if it is prohibited to disclose their family names in association with the web-sites they visited then it is also prohibited to disclose their names in association with the web-sites they visited.

We have another interesting property connecting the notions of permitted and obligatory communicative acts. Let $\phi, \phi' \in \mathcal{L}^{\phi}_{DEDL}$:

$$\text{If} \vdash \phi' \to \phi \text{ then} \vdash O_s(send\ \phi') \to \neg P_s(send\ \neg\phi)$$

This proposition states that if it is obligatory to disclose a fact then it is prohibited to disclose the opposite of any of its logical consequences. However, note that $O_s(send\ \phi)$ and $P_s(send\ \phi)$ are not dual operators:

$$\nvdash O_s(send\ \phi) \leftrightarrow \neg P_s(send\ \neg\phi).$$

This is intuitively correct: in Example 9 it is prohibited to disclose e but it does not entail that it is obligatory to disclose $\neg e$. Moreover, we have the following property:

$$\nvdash P_s(send\ \phi) \wedge P_s(send\ \psi) \to P_s(send\ (\phi \wedge \psi)).$$

Indeed, in Example 8 we had $M, w \models P_s(send\ e) \wedge P_s(send\ c) \wedge \neg P_s(send\ (e \wedge c))$.

Enforcing privacy policies: $[\boldsymbol{prom\ \phi}]$. The hierarchical superior of the *sender* or the *sender* himself might decide to change the policy privacy from $\mathcal{P}$ to $\mathcal{P}'$. As a result, the sender needs to enforce this new privacy policy $\mathcal{P}'$. This enforcement is captured in our formalism by $[prom\ \psi]$.

Example 11. (Website Example) In case of attack by some hacker, the privacy policies can be made more strict. For example, the *sender* can decide to strengthen the privacy policy $\mathcal{P}_1$ of Example 6 to

$$\mathcal{P}_4 = \{P_s K'_r c, \neg P_s K'_r e, \neg P_s K'_r v\}$$

where $P_s K'_r e$ has been replaced by $\neg P_s K'_r e$: it is now prohibited to disclose the mapping from numbers to visited websites. This new privacy policy $\mathcal{P}_4$ can be enforced by the *sender* through the update $[prom\ \neg K'_r e]$. We get the EDL-model $(M * \neg K'_r e, w)$ depicted in Figure 4 which is compliant with respect to $\mathcal{P}_4$.

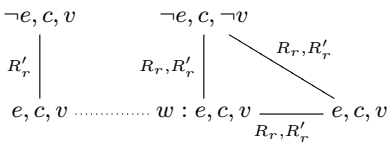

Fig. 4. Website example updated

4 Checking compliance and changing policies

The general language $\mathcal{L}_{DEDL}$ we defined is not completely appropriate for a security monitor (the *sender*) to reason about a situation given a privacy policy. Indeed, it does not allow him to express that the situation is compliant or not with respect to the privacy policy. It does not allow him to express that there is a change of privacy policy and that the new privacy policy is now $\mathcal{P}'$. It does not allow him to plan actions so that the new privacy policy is enforced. It does not allow him to express that the current privacy policy is $\mathcal{P}$ and that under this privacy policy he is permitted to disclose ϕ. These kinds of statements are needed if we want the decision maker to be able to enforce and maintain a privacy policy. So we need to define a new language based on $\mathcal{L}_{DEDL}$ more appropriate in the context of privacy regulation. This language should allow the decision maker to refer explicitly to the current privacy policy which was always implicitly present in the previous language. So we propose the following language $\mathcal{L}_{PL}$ whose formulas are denoted ϕ^*:

$$\mathcal{L}^{\phi}_{PL} : \phi ::= p \mid \mathcal{P} \mid c \mid \neg\phi \mid \phi \wedge \phi \mid K_r\phi \mid O_s\alpha \mid P_s(send\ \phi) \mid$$
$$[send\ \phi]\phi \mid [learns\ \mathcal{P}]\phi \mid [prom\ \alpha]\phi$$
$$\mathcal{L}^{\alpha}_{PL} : \alpha ::= K'_r\phi \mid \neg\alpha \mid \alpha \wedge \alpha \mid$$
$$[send\ \phi]\alpha \mid [learns\ \mathcal{P}]\alpha \mid [prom\ \alpha]\alpha$$

where p ranges over Φ^{ϕ} and $\mathcal{P}$ over the set of privacy policies Pol. We assume here that the set of privacy policies Pol is finite and that each of them has a finite number of epistemic norms.

So we have five new kinds of formulas referring each of them directly or indirectly to privacy policies: $\mathcal{P}$, $[learns\ \mathcal{P}]\phi$, $P_s(send\ \psi)$ and c. $\mathcal{P}$ reads 'the privacy policy is $\mathcal{P}$'. $[learns\ \mathcal{P}]\phi$ reads 'after *sender* learns that the new privacy policy is $\mathcal{P}$, ϕ holds'. $P_s(send\ \phi)$ reads 'sending the message ϕ is permitted'. c reads 'the situation is compliant w.r.t. $\mathcal{P}$'. This language allows to express all the new kinds of statement we wanted to express above. For example, $\mathcal{P} \wedge \neg c$ means that the current privacy policy is $\mathcal{P}$ but the current situation is not compliant with this privacy policy. The formula $\neg c \wedge [send\ \phi]c$ means that the current situation is not compliant with respect to the privacy policy but if ϕ is disclosed then the situation becomes compliant with this privacy policy. The formula $\mathcal{P} \wedge \neg c(\mathcal{P})$ means that the privacy policy is $\mathcal{P}$ but this privacy policy is not enforced.

The semantics for this language is a bit different from the semantics of $\mathcal{L}_{DEDL}$ because we have to refer explicitly in the language to privacy policies. Intuitively,

$\{(M,w),\mathcal{P}\}$ in the definition below is the situation (M,w) where the privacy policy $\mathcal{P}$ holds.

Definition 7. *A (pointed) privacy model, is a pair $\{M,\mathcal{P}\}$ (resp. $\{(M,w),\mathcal{P}\}$) composed of an EDL-model M (resp. (M,w)) together with a privacy policy $\mathcal{P}$. The truth conditions are defined inductively as follows:*

$$
\begin{array}{lll}
\{(M,w),\mathcal{P}\} \models p & \textit{iff} & M,w \models p \\
\{(M,w),\mathcal{P}\} \models \mathcal{P}' & \textit{iff} & \models c(\mathcal{P}) \leftrightarrow c(\mathcal{P}') \\
\{(M,w),\mathcal{P}\} \models c & \textit{iff} & M,w \models c(\mathcal{P}) \wedge \bigwedge\limits_{\phi\in\mathcal{P}} (\phi \rightarrow t(\alpha_\phi)) \\
\{(M,w),\mathcal{P}\} \models P_s(send\ \phi) & \textit{iff} & \{(M,w),\mathcal{P}\} \models [send\ \phi]c \\
\{(M,w),\mathcal{P}\} \models [learns\ \mathcal{P}']\phi & \textit{iff} & \{(M,w),\mathcal{P}'\} \models \phi \\
\{(M,w),\mathcal{P}\} \models [send\ \phi]\phi^* & \textit{iff} & \{(M*\phi,w),\mathcal{P}\} \models \phi^* \\
\{(M,w),\mathcal{P}\} \models [prom\ \alpha]\phi^* & \textit{iff} & \{(M*\alpha,w),\mathcal{P}\} \models \phi^*
\end{array}
$$

The other inductive steps are like in Definitions 1 and 8 except that M,w has to be replaced here by $\{(M,w),\mathcal{P}\}$.

In the fourth truth condition, there is not necessarily a logical relation between $\mathcal{P}$ and $\mathcal{P}'$ since the privacy policy can change for any reason. The second truth condition entails that we do not distinguish privacy policies if they are compliant in exactly the same situations.

Theorem 3. *The semantics of $\mathcal{L}_{PL}$ is* sound and complete *with respect to the (privacy) logic PL axiomatized by the following axiom schemes and inference rules. PL is also* decidable.

$$
\begin{array}{ll}
L_{DEDL} & \textit{All the axioms schemes and inference rules of } L_{DEDL} \\
P_0 & \vdash c \bigwedge\limits_{\mathcal{P}\in Pol} \left(\mathcal{P} \rightarrow \left(c(\mathcal{P}) \wedge \bigwedge\limits_{\phi\in\mathcal{P}} (\phi \rightarrow t(\alpha_\phi))\right)\right) \\
P_1 & \vdash \mathcal{P} \leftrightarrow \mathcal{P}' \textit{ iff } \vdash c(\mathcal{P}) \leftrightarrow c(\mathcal{P}') \\
P_2 & \vdash P_s(send\ \phi) \leftrightarrow [send\ \phi]c \\
P_3 & \vdash [send\ \phi]\mathcal{P} \leftrightarrow \mathcal{P} \\
P_4 & \vdash [prom\ \phi]\mathcal{P} \leftrightarrow \mathcal{P} \\
P_5 & \vdash [learns\ \mathcal{P}]p \leftrightarrow p \\
P_6 & \vdash [learns\ \mathcal{P}]\mathcal{P} \\
P_7 & \vdash [learns\ \mathcal{P}]\neg\phi \leftrightarrow \neg[learns\ \mathcal{P}]\phi \\
P_8 & \vdash [learns\ \mathcal{P}]K_r\phi \leftrightarrow K_r[learns\ \mathcal{P}]\phi \\
P_9 & \vdash [learns\ \mathcal{P}]O_s\alpha \leftrightarrow O_s[learns\ \mathcal{P}]\alpha \\
P_{10} & \vdash [learns\ \mathcal{P}]K'_r\phi \leftrightarrow K'_r[learns\ \mathcal{P}]\phi \\
P_{11} & \vdash [learns\ \mathcal{P}](\phi \rightarrow \phi') \\
 & \rightarrow ([learns\ \mathcal{P}]\phi \rightarrow [learns\ \mathcal{P}]\phi') \\
R_P & \textit{If} \vdash \phi^* \textit{ then} \vdash [learns\ \mathcal{P}]\phi^*
\end{array}
$$

Proof (Proof sketch). We use the same method as for the proof of Theorem **??**. P_2 to P_{11} and R_P allow to reduce a formula of $\mathcal{L}_{PL}$ to a formula of $\mathcal{L}_{PL}$ without dynamic operators. We build the canonical model for this restricted language completely similarly to the canonical model M of Theorem 1 and we set $M, \Gamma \models \mathcal{P}$ if $\mathcal{P} \in \Gamma$. P_1 then ensures that Condition (2) of Definition 7 is fulfilled. Decidability is proved using the same method as for the proof of Theorem **??**.

Example 12. The mechanisms involved in the website example can be better analysed and understood with this new language. In Example 6, the privacy policy is $\mathcal{P}_1$ and the initial situation is compliant w.r.t. this privacy policy:

$$\{(M, w), \mathcal{P}_1\} \models c \wedge \mathcal{P}_1.$$

After the *sender* learns that the new privacy policy is $\mathcal{P}_4$, the situation is no longer compliant with this new privacy policy because the privacy policy $\mathcal{P}_4$ is not enforced anymore:

$$\{(M, w), \mathcal{P}_1\} \models [learns\ \mathcal{P}_4]\left(\neg c \wedge (\mathcal{P}_4 \wedge c(\neg \mathcal{P}_4))\right).$$

In that case, we reach a non-compliant situation $\{(M, w), \mathcal{P}_4\}$ because we have $\{(M, w), \mathcal{P}_4\} \models \mathcal{P}_4 \wedge \neg c(\mathcal{P}_4)$. Therefore, *sender* now has to enforce this new privacy policy $\mathcal{P}_4$ by means of a promulgation. He does so by promulgating the norm $\neg K'_r e$. That was the process described in Example **??**:

$$\{(M, w), \mathcal{P}_4\} \models \neg c \wedge [prom\ \neg K'_r e]c.$$

We see in the above example that the language $\mathcal{L}_{PL}$ really allows the security monitor to reason about which actions he can perform so that a new privacy policy be enforced or so that the situation be compliant w.r.t. the privacy policy.

5 Conclusion

Related work. Languages for access control in security have been used for modelling privacy regulations too [8]. However, they are not easily adapted to the new task, for example, because they do not provide ways of reasoning about the information and about effects of messages. Moreover, they rarely consider the context of communication.

Specific languages for privacy policies have been proposed, but have some limitations. Extensible Access Control Markup Language XACML's policies can lead to obligations, but "obligation" is just an uninterpreted symbol which receives meaning at the point of policy enforcement [2]. Enterprise Privacy Authorization Language EPAL's policies are concerned with a single sender (the enterprise itself) and a single recipient role, like in our model [19]. EPAL structures obligations with a subsumption relation rather than allowing to reason about knowledge like us. The Platform for Privacy Preferences (P3P) language contains only positive norms and very restricted temporal conditions [13].

Cuppens and Demolombe [16] extends the original framework [14] by using an epistemic deontic logic to model security in databases. They do not introduce dynamics

in their system, neither for knowledge nor for deontic operators, even if they recognize the importance of this aspect. We share many properties of their epistemic-deontic modalities, but we also extend them to permissions and obligations concerning actions and not only propositions, getting a more fine grained analysis, for example of the Chinese wall problem. Moreover, they do not introduce separately the epistemic and deontic operators but only combined ones, like [10] do, limiting the expressivity of the logic. Our modularity allows us to model more complex formulas which can express meta-security policies or obligations to know whether something holds. Given that our approach is based on their approach, their solutions to several problems can naturally be transferred in our setting. They show for example that multi-level security policies which assign a degree of clearance l to formulae ϕ and which might be incomplete can be expressed in their framework by indexing the modality $P_s K'_r \phi$ with the degree of clearance l: $P_s K'_{rl} \phi$ reads 'an agent r cleared at level l is explicitly permitted to know that the database believes ϕ'. They also avoid possible conflicts between roles and regulations by defining the role of an agent as an index i of the modality $P_s K'_{ri} \phi$ and by introducing an external structure on these roles.

Bonatti *et al.* [10] use a similar logical framework for reasoning about security in database access: they explicitly model the beliefs of the user of the database and the actions which change these beliefs. However, they do not make an explicit distinction between epistemic and deontic modalities, with resulting limitations such as the impossibility to model permissions and obligations about actions. Moreover, the belief change mechanism is superimposed to Kripke semantics, while we use a general epistemic dynamic logic approach and we are also able to change permissions and obligations and not only beliefs. As they do, by distinguishing the point of view of the database (*sender*) from the beliefs of the user (*recipient*), we could model situations where the sender of information is lying, even if this possibility seems less useful in the context of privacy regulations. Finally, we can model meta-security in our framework, as proposed by the authors, to specify that it is prohibited to know the privacy policy. Differently from their work, we can provide also a semantics to meta-security since we allow nestings of epistemic and deontic modalities.

Barth *et al.* [6] propose a formalization of the theory of privacy called contextual integrity. They introduce positive and negative norms, depending on whether they refer to actions that are allowed or disallowed. Temporal conditions are modelled by means of linear temporal logic with past and future operators to express, for example, that certain information may be disclosed only if the subject mentioned has previously given permission or that if certain information is made public, notification must be sent to the concerned party. These norms are interpreted in a model of agents who respect the norms if the trace history of their communication satisfies a temporal formula constructed from the norms by taking the disjunction over positive norms and the conjunction over negative norms. Their language constitute an advancement with respect to other policy languages, both for the temporal aspect and for including a relation enabling agents to combine messages to compute additional information about the subject, (e.g., computing postal code from postal address), elucidating the notion of a "data hierarchy" found in P3P and EPAL. However, their privacy policies cannot be changed. On the other hand, we do not consider the temporal aspect yet: to incorporate this aspect in

our model it might be necessary to resort to an epistemic temporal logic, as in Pacuit and Parikh [26]. However, in [26], only particular epistemic norms called knowledge-based obligations of the form $K_r\phi \rightarrow O_s\psi$, where ψ does not contain any knowledge operator, can be expressed.

A problem of Barth *et al.* [6] is the obscurity of the formalism used to model legal norms, which in turn present ambiguities and difficulties. To cope with this problem [21] propose a more readable formalism based on logic programming. Our modal logic aims at improving readability too, but at the same time it allows to study precisely the properties of the deontic operators.

Logic or logic programming (see also [5]) are not the only methodologies to formalize privacy regulations. A recent example is [22] where they use an extension of access control matrix operations to include operations for notification and logging and constructs that ease the mapping between legal and formal language. They apply their methodology to HIPAA regulations of health insurance. [25] proposes to use π-calculus for privacy in the context of service oriented architectures.

A further issue in privacy is the interaction between policies and the organizations which have to enforce them. This is addressed, e.g., by [7] and [18]. Our plan to address this problem is to extend the modal language to a multi-agent language in order to express obligations, beliefs, knowledge and *goals* of the different parties involved.

In dynamic epistemic logic, [4] is the closest work to ours. They focus in a multi-agent setting on the notion of permission to announce. They provide a sound, complete and decidable logic by enriching public announcement logic with the operator $P(\psi, \phi)$ which reads 'after ψ has been publicly announced, it is permitted to say ϕ'. There is no real notion of privacy policy nor compliance, although the specification of such a policy could be somehow derived via the specification of their operator $P(\psi, \phi)$ (whose first argument handles the dynamic character of the situations they consider). But as in all the other approaches mentioned, the (implicit) privacy policy is specified directly on the announcements/actions and the epistemic character of the situations they consider does not really play a role. Finally, in their logic, privacy policies cannot change and they do not have a notion of obligatory announcement or enforcement (although such issues are addressed independently at the end of their paper).

Concluding remarks. In this paper, we introduced a logic satisfying the four requirements of the introduction. In order to use this logic in real situations, the security monitor (*sender*) would need to implement an EDL-model representing the current epistemic/deontic state of affairs. He could then check compliance w.r.t. a given policy and determine which actions can and should be done by model checking this EDL-model.

A topic for further research is to deal with multi-agent scenarios involving more agents than just a *sender* and a *recipient*, each agent having its own privacy policy. Another topic for further research is to enrich the dynamics to allow not only operations which add new regulations but also operations which remove or revise regulations.

References

1. C. Alchourrón and P. Gärdenfors and D. Makinson. On the Logic of Theory Change: Partial Meet Contraction and Revision Functions. *Journal of Symbolic logic*, 50(2):510–530, 1985.

2. A. Anderson *et al.* Extensible access control markup language (XACML) version 2.0. 2004.
3. G. Aucher. A Combined System for Update Logic and Belief Revision. Masters thesis. ILLC, University of Amsterdam, the Netherlands. 2003.
4. P. Balbiani, H. van Ditmarsch, and P. Seban. Reasoning about permitted announcements. In *ESSLLI 2009 workshop Logical Methods for Social Concepts*, Bordeaux, 2009.
5. S. Barker. Protecting deductive databases from unauthorized retrieval and update requests. *Data and Knowledge Engineering*, 43(3):293-315, 2002.
6. A. Barth, A. Datta, J. C. Mitchell, and H. Nissenbaum. Privacy and contextual integrity: Framework and applications. In *19th IEEE Symposium on Security and Privacy*, pages 184–198. IEEE Computer Society, 2006.
7. A. Barth, J. C. Mitchell, A. Datta and S. Sundaram. Privacy and contextual integrity: Framework and applications. In *20th IEEE Computer Security Foundations Symposium, CSF 2007*, pages 279-294. IEEE Computer Society, 2007.
8. M. Bishop. *Computer Security: Art and Science*. Addison Wesley Professional, 2003.
9. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Computer Science*. Cambridge University Press, 2001.
10. P. Bonatti, S. Kraus and V. Subrahmanian. Foundations of Secure Deductive Databases. *IEEE Transactions on Knowledge Data and Engineering*, 7(3):406–422, 1995.
11. H.-N. Castañeda. *The paradoxes of Deontic Logic: the simplest solution to all of them in one fell swoop*, pages 37–86. Synthese library. 1981.
12. H.-N. Castañeda. Knowledge and epistemic obligation. *Philosophical perspectives*, 2:211–233, 1988.
13. L. Cranor. *Web Privacy with P3P*. O'Reilly and Associates Inc., 2002.
14. F. Cuppens. A Logical Formalization of Secrecy. In *6th IEEE Computer Security Foundations Workshop - CSFW'93*. IEEE Computer Society, 1993.
15. F. Cuppens and R. Demolombe. Normative Conflicts in a Confidentiality Policy. In *ECAI Workshop on Artificial Normative Reasoning*. 1994.
16. F. Cuppens and R. Demolombe. A Deontic Logic for Reasoning about Confidentiality. In *Deontic Logic, Agency and Normative Systems, DEON '96: Third International Workshop on Deontic Logic in Computer Science*. Springer, 1996.
17. F. Cuppens and R. Demolombe. A Modal Logical Framework for Security Policies. In *Foundations of Intelligent Systems, 10th International Symposium, ISMIS '97*. Springer, pages 579-589, 1997.
18. M. Kanovich, P. Rowe and A. Scedrov. Collaborative Planning With Privacy. In *20th IEEE Computer Security Foundations Symposium, CSF 2007*. pages 265-278, 2007.
19. G. Karjoth and M. Schunter. A privacy policy model for enterprises. In *15th IEEE Computer Security Foundations Workshop*. IEEE Computer Society, 2002.
20. B. Kooi. Probabilistic dynamic epistemic logic. *Journal of Logic, Language and Information*, 12(4):381–408, 2003.
21. P. Lam, J. Mitchell and S. Sundaram. A Formalization of HIPAA for a Medical Messaging System. In *Trust, Privacy and Security in Digital Business, TrustBus 2009*.
22. M. May, C. Gunter and I. Lee. Privacy APIs: Access Control Techniques to Analyze and Verify Legal Privacy Policies. In *19th IEEE Computer Security Foundations Symposium CSFW-19*, pages 85-97, 2006.
23. R. van der Meyden. The Dynamic Logic of Permission. *Journal of Logic and Computation*, 6(3):465-479, 1996.
24. J.J. Meyer. A Different Approach to Deontic Logic: Deontic Logic Viewed as a Variant of Dynamic Logic. *Notre Dame Journal of Formal Logic*, 29(1):109-136, 1988.
25. H. Nielson and F. Nielson. A flow-sensitive analysis of privacy properties. In *20th IEEE Computer Security Foundations Symposium CSFW'07*, pages 249-264, 2007.

26. E. Pacuit and R. Parikh. The logic of knowledge based obligation. *Synthese*, 149(2), 2006.
27. H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic*, volume 337 of *Synthese library*. Springer, 2007.

A An extension of Castañeda's deontic logic

In this appendix, we give an extension of our epistemic deontic logic which embeds Castañeda's deontic logic. Starting from a linguistic analysis, the insight of Castañeda is to acknowledge the grammatical duality of expressions depending on whether they are within or without the scope of an obligation operator. This leads him formally to introduce two sets of formulas: circumstances which cannot *alone* be the foci of deontic operators, unlike what he calls practitions. The former are usually expressed grammatically in the indicative form and the latter are usually expressed grammatically in the infinitive/subjunctive form. For example, "Freud cures Anna O" in the indicative form is a circumstance, but the same sentence in "*it is obligatory* that Freud cures Anna O" in subjunctive/infinitive form is a practition. Just as practitions are the foci of deontic operators, circumstances are dually the foci of knowledge operators, as pointed out by Castañeda [12]. Note that an expression ϕ in the scope of a knowledge operator $K_r\phi$ is always in the indicative form and never in the subjunctive/infinitive form, even if $K_r\phi$ is in the scope of a deontic operator O.

We extend Castañeda [12]'s intuition to the context of epistemic permissions and obligations. In a deontic setting the reading of the term knowledge or belief can be twofold: either as a circumstance or as a practition. On the one hand, in the sentence "it is obligatory that John *knows* / for John *to know* that there is an infinity of prime numbers" the verb 'to know' is the focus of a deontic operator and is in the subjunctive/infinitive form. On the other hand, the sentence "John *knows* that there is an infinity of prime numbers" alone describes a circumstance and the interpretation of the verb 'to know' in the indicative form matches the one usually studied in epistemic logic. The former use of the term knowledge within the scope of a deontic operator is not studied in epistemic logic. For these reasons we enrich the language of Castañeda with two knowledge modalities, one for circumstances and the other one for epistemic practitions. This yields the following language $\mathcal{L}'_{EDL} = \mathcal{L}^{\phi'}_{EDL} \cup \mathcal{L}^{\alpha'}_{EDL}$:

$$\mathcal{L}^{\phi'}_{EDL} : \phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid K_r\phi \mid O_s\alpha$$
$$\mathcal{L}^{\alpha'}_{EDL} : \alpha ::= \beta \mid K'_r\phi \mid \neg\alpha \mid \alpha \wedge \alpha \mid \alpha \wedge \phi \mid \phi \wedge \alpha$$

where p ranges over Φ^ϕ and β over Φ^α. The only difference with the language $\mathcal{L}_{EDL}$ is that we now have pure practitions Φ^α and that practitions can now be of the form $\phi \wedge \alpha$ or $\phi \to \alpha$ where ϕ is a proposition. Pure practitions Φ^α are expressions in the scope of a deontic operator that cannot be expressed with a knowledge operator, such as 'to cure Anna O' in 'it is obligatory to cure Anna O'. Therefore, just as epistemic practitions, they are in the subjunctive/infinitive form. Moreover, with this definition of practitions we can also express formulas of the form $O_s(\phi \to \alpha)$. Obviously, we would like to have the following validity:

$$\models O_s(\phi \to \alpha) \leftrightarrow (\phi \to O_s\alpha)$$

which is a generalization to the epistemic case of Castañeda's key validity. For example, "it is obligatory that if Freud knows that Anna O is sick, then he cures her" ($O_s(K_r\phi \rightarrow \alpha)$) is intuitively equivalent to "if Freud knows that Anna O is sick, then it is obligatory that he cures her" ($K_r\phi \rightarrow O_s\alpha$). To obtain this validity, we need to add an extra condition $(*)$ in our definition of EDL-model and so define EDL-model'.

Definition 8. *An EDL-model' M is a tuple $M = (W, D, R_r, R'_r, V)$, where W is a non-empty set of possible worlds, R_r, R'_r and D are accessibility relations on W, D being serial, and V is a valuation such that:*

for all $w \in W$, all $v, v' \in D(w) \cup \{w\}$, (M, v) is R_rD-bisimilar to (M, v'). $(*)$

The semantic condition $(*)$ intuitively means that the (epistemic) context where a normative system applies is fixed. One can easily show that any Castañeda model [11] can be embedded into an EDL-model', in the sense that the Castañeda model and the corresponding EDL-model' satisfy the same formulas of $\mathcal{L}'_{EDL}$ without epistemic operators K_r or K'_r. One can also show that the semantics of $\mathcal{L}'_{EDL}$ is *sound and complete* with respect to the logic L_{EDL} to which we add the axiom scheme $\vdash O_s(\phi \rightarrow \alpha) \leftrightarrow (\phi \rightarrow O_s\alpha)$. In this new *decidable* logic, we can then derive the theorem $\vdash O_sK'_r\phi \rightarrow \phi$.