TRANSACTIONS ON DATA PRIVACY 7 (2014) 223–252

Censors for Boolean Description Logic

Thomas Studer, Johannes Werner

Universität Bern, Institut für Informatik und angewandte Mathematik, Neubrückstr. 10, 3012 Bern, Switzerland

E-mail: {tstuder,werner}@iam.unibe.ch

Abstract. Protecting different kinds of information has become an important area of research. One aspect is to provide effective means to avoid that secrets can be deduced from the answers of legitimate queries. In the context of atomic propositional databases several methods have been developed to achieve this goal. However, in those databases it is not possible to formalize structural information. Also they are quite restrictive with respect to the specification of secrets. In this paper we extend those methods to match the much greater expressive power of *Boolean description logics*. In addition to the formal framework, we provide a discussion of various kinds of censors and establish different levels of security they can provide.

Keywords: Information security, knowledge-base system, data privacy, description logic, query based evaluation

1 Introduction

In the past decades the possibilities to combine and exchange information aggrandised dramatically. At the same time also the speed of this exchange increased by several orders of magnitude. Unfortunately, with this development came a huge loss of privacy.

Because of the high computational power of modern computers, databases, and information networks, it is not sufficient to only consider privacy violations caused by information that is stored in a database or that is an immediate consequence of stored data. We also have to consider that secret information may be logically inferred (even indirectly or via meta-reasoning) from answers provided by a database. The stored and communicated information can, e.g. by inference, lead to *supposed* knowledge or belief, which can cause serious problems, e.g. lost or unawarded jobs, wrong lawful accusations, high insurance fees, etc.

However, in most cases, it is well known what information must or should be considered harmful, at the very least to the person or system protecting the information. Hence those secrets should not be believed or known by any untrusted human or artificial agent querying the data stock.

This need, to protect some information from being revealed, is opposed by the need to be certain about some related information. For instance in healthcare it is necessary to have information about the spread of an infection and possible infection zones, but not desirable to give away names (or other identifying data) of infected people to avoid, e.g., any harassment.

Hence it is necessary to develop techniques that ensure non-revealing of potentially dangerous information, but allow to make public other information whenever it is safe. Query based evaluation is a successful approach for privacy preserving query answering. The basic idea of this method is to distort the answer to a query if it would leak sensitive information. This technique to preserve privacy has been developed in [6] for (Boolean, atomic) complete databases and in [10] for incomplete databases.

However, often not only atomic facts (like "has cancer" or "is alcoholic") must be protected, but also structural information (like "at least one of my co-workers has a contagious disease"). To this end we extend in this paper the approach of the Boolean incomplete case to a logic capable of representing structural information, namely Boolean $\mathcal{ALC}$. This modification of the description logic $\mathcal{ALC}$ (compare [2]) supports Boolean combinations of subsumption formulae. While not being expressive enough to handle any structural information, it provides a good perspective on the area to be explored. Furthermore we drop the atomicity property usually assumed in databases, i.e. that it is sufficient to evaluate propositional letters to either *true* (t), *false* (f), or *unknown* (u).

The key idea pioneered in [6] and [10] is to equip the database with a so-called censor. The task of this censor is to intermediate between an actual database and a querying agent in such a way that data privacy is guaranteed. To achieve this, the censor has, for instance, the ability to lie, to refuse an answer (that is to answer *refuse* (r)), and to recall previous queries and its own corresponding answers. This separates the task of maintaining privacy from the task of data keeping and thus provides more flexibility in the protective measures than an integrated approach (e.g. hidden rows in the database).

Outline of the paper. We start with recalling the semantics of Boolean $\mathcal{ALC}$ and define what it means to evaluate a query over a knowlege base. Further, we need a notion of knowledge to talk about what an attacker knows and what the attackers considers possible. Biskup and Weibert [10] approach this by using a suitable embedding of the privacy problem into the modal logic S5. However, it is not possible to utilize an S5-like structure directly in the case of description logics. Thus we will adapt the embedding procedure and keep only the essential parts. More precisely, we will only use a structure reflecting one main connected component of the S5-structure, which represents the heart and muscle of the method. Since such a component is fully described by its members, we drop the equivalence relation and restrict our view to a *cloud*, i.e. a set of interpretations (models).

Making use of the so obtained framework, we will formally introduce the relevant concepts for censor-based query answering in $\mathcal{ALC}$ knowledge-bases. In particular we will provide definitions for the following notions.

Privacy configuration. A privacy configuration is a triple that consists of the attacker's knowledge, the knowledge-base that can be queried, and the set of secrets that should not be revealed to the attacker.

Censor. A censor provides an answering function for each privacy configuration. This answering function may distort the answer to a query if the correct answer would leak sensitive information.

Credible. A censor is credible if its answers do not contradict each other, that is if they provide a consistent view to the attacker.

Effective. A censor is effective if it keeps all secrets.

Continuous. This is a technical notion saying that the answers a censor provides only depend on previous queries and answers (but not on future ones).

Truthful. A censor is truthful if it does never lie. But it may refuse to answer a query.

Lying. A censor is lying if it may give incorrect answers.

Minimally invasive. A censor is minimally invasive if it distorts an answer only if otherwise a secret would be leaked.

Repudiating. A censor is repudiating if for any sequence of queries, there is an alternate knowledge-base in which all secrets are not true and which, given as input to the censor, would produce the same answers as the ones computed from the actual knowledge-base. Hence a repudiating censor can plausibly deny all secrets even if the algorithm of the censor is known to the attacker.

We first establish some basic properties of these notions. For instance, we have:

1. truthful implies credible,
2. effective implies credible if the set of secrets is non-empty.

We proceed with studying truthful censors. In particular, we present algorithms for two censors:

1. one that is credible, effective, truthful, continuous, and repudiating but not minimally invasive;
2. one that is credible, effective, truthful, continuous, and minimally invasive but not repudiating.

These two censors are optimal. Namely, we show that a continuous truthful censor cannot simultaneously be credible, minimally invasive and repudiating.

Next we study lying (but not refusing) censors. We present an algorithm for a censor that is credible, effective, lying (but not refusing), continuous, repudiating, and minimally invasive. Hence, in contrast to truthful censor, we can simultaneously have all desired properties for lying censors. Therefore, there is no need to consider censors that combine both lying and refusing.

Last but not least we present a censor that is credible and repudiating but not effective. Thus (maybe surprisingly) repudiating does not imply effective.

Related work. In a controlled query evaluation approach to privacy, the answer to a query is distorted if otherwise it would leak sensitive information to the user. As mentioned before, there are basically two distortion methods: the answer can simply be refused [19] or the system can give an incorrect answer (that is it lies) [11]. The framework of controlled query evaluation has been applied for a variety of data models and control mechanisms, see for instance [7, 8, 9, 10].

The work that is probably most closely related to ours is [9] by Biskup and Bonatti who study controlled query evaluation for a decidable relational submodel. Their results are very general and include, for instance, even the Bernays–Schönfinkel fragment of classical first-order logic. However, they rely on the *closed world assumption*, which is standard for relational systems. Our work, on the other hand, uses the *open world assumption*, which is standard for description logics and knowledge base systems like ontological information

systems. The difference between open and closed world assumption is the way negation is treated, see, e.g., [2]. Suppose we have a knowledge base $\mathcal{KB}$ that stores information about a person called Peter. Consider a proposition `isMarried` and assume

$$\texttt{isMarried} \notin \mathcal{KB}. \tag{1}$$

Using the closed world assumption, (1) means that Peter is not married. Using the open world assumption, (1) means that it is not known whether Peter is married. This is the approach we take in this paper. Hence in our setting issuing the query `isMarried` against the knowledge base $\mathcal{KB}$ will produce the answer `unknown`.

Although knowledge-base systems enter more and more application domains, privacy issues in the context of description logics are not yet well studied. Notable exceptions are the following. Calvanese et al. [12] address the problem of privacy aware access to ontologies. They show how view based query answering is able to conceal from the user information that are not logical consequences of the associated authorization views. Based on this approach, Grau et al. [14] recently presented a controlled query evaluation framework for lightweight ontologies.

Grau and Horrocks [13] study different notions of privacy for logic based information systems. Their idea is to look at privacy preserving query answering as reasoning problems. They establish a connection between these reasoning problems and probabilistic privacy guarantees such as perfect privacy [18]. Bao et al. [5] present a safe reasoning strategy for the description logic SHIQ and for hierarchical ontologies. Their approach is based on the concepts of locality and conservative extensions for description logics. A complete decision procedure for provable data privacy [20] in the context of $\mathcal{ALC}$ is introduced in [21]. Privacy preserving query answering over modular ontologies that are given in the very expressive description logic SHOIQ is examined in [22].

Boolean description logics, i.e. description logics that feature Boolean combinations of subsumption statements $C \sqsubseteq D$ and/or assertional statements $C(a)$ have been studied in many forms and various contexts, see, for instance, [1, 3, 15, 16, 17, 23].

2 Definitions

2.1 Syntax

Despite the fact that $\mathcal{ALC}$ usually denotes only satisfiability of conceptual knowledge [2], namely T-Boxes, i.e. sets of subsumption statements, and A-Boxes, i.e. (positive) assertional statements about individuals, throughout this paper we will refer to Boolean $\mathcal{ALC}$ as $\mathcal{ALC}$. All formulae in this paper are hence Boolean combinations of subsumption statements.

Definition 1 ($\mathcal{ALC}$ and $\mathcal{CALC}$)**.** Given two disjoint sets of symbols $\mathscr{HC}$ (atomic concepts) and $\mathscr{HR}$ (atomic roles), the languages of $\mathcal{ALC}$ and $\mathcal{CALC}$ are defined by the following grammar in Backus-Naur form:

$$\begin{aligned}
\Phi &::= \Box\psi \mid \Diamond\psi \\
\psi &::= t \mid f \mid \psi \wedge \psi \mid \neg\psi \mid C \sqsubseteq C \\
C &::= C_i \mid \bot \mid \top \mid C \sqcap C \mid \overline{C} \mid \exists R.C \mid \forall R.C \\
R &::= R_i
\end{aligned}$$

Here $C_i \in \mathscr{HC}$ are the atomic concepts and $R_i \in \mathscr{HR}$ are atomic roles (or role names). By this BNF we define the sets $\mathcal{R}$ (Roles), $\mathcal{C}$ (Concepts), $\mathcal{L}_{\mathcal{ALC}}$ ($\mathcal{ALC}$-formulae) and $\mathcal{L}_{\mathcal{CALC}}$

($\mathcal{CALC}$-formulae) as the sets of words that can be derived starting from R, C, ψ and Φ respectively. Further we refer to a set of $\mathcal{ALC}$-formulae as *knowledge-base* and to the pair $(\mathscr{AR}, \mathscr{AC})$ as its (*description*) *basis*.

For the sake of simplicity we use the standard abbreviations like $\rightarrow, \vee, \sqsupseteq, \equiv$ and so on. Let us point out, that on the level of $\mathcal{L}_{\mathcal{CALC}}$ there are no logical (binary) connectives. The formulae of $\mathcal{L}_{\mathcal{CALC}}$ are just the formulae of $\mathcal{L}_{\mathcal{ALC}}$ prefixed by diamond ($\Diamond$) or box ($\Box$).

2.2 Semantics

2.2.1 $\mathcal{ALC}$-Models

Definition 2 ($\mathcal{ALC}$-interpretation)**.** Given a description basis $(\mathscr{AR}, \mathscr{AC})$, an ($\mathcal{ALC}$-) *interpretation* is a pair $(\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$, consisting of a non-empty *domain* $\Delta_{\mathcal{I}}$ and a function

$$\cdot^{\mathcal{I}} : \mathcal{C} \cup \mathcal{R} \rightarrow \wp(\Delta_{\mathcal{I}}) \cup \wp(\Delta_{\mathcal{I}} \times \Delta_{\mathcal{I}})$$

that satisfies the following conditions:

- $\top^{\mathcal{I}} = \Delta_{\mathcal{I}}$, $\bot^{\mathcal{I}} = \emptyset$
- for each atomic concept $A \in \mathscr{AC}$: $A^{\mathcal{I}} \subseteq \Delta_{\mathcal{I}}$
- for each atomic role $R \in \mathscr{AR}$: $R^{\mathcal{I}} \subseteq \Delta_{\mathcal{I}} \times \Delta_{\mathcal{I}}$
- for each compound concept it inductively holds
  - $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
  - $(\overline{C})^{\mathcal{I}} = \Delta_{\mathcal{I}} \setminus C^{\mathcal{I}}$
  - $(\exists R.C)^{\mathcal{I}} = \{a \in \Delta_{\mathcal{I}} \mid \exists b \in C^{\mathcal{I}} : (a, b) \in R^{\mathcal{I}}\}$
  - $(\forall R.C)^{\mathcal{I}} = \{a \in \Delta_{\mathcal{I}} \mid \forall b \in \Delta_{\mathcal{I}} : (a, b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$

At this stage the constants t and f are superfluous and could be replaced by defining $t := \bot \sqsubseteq \top$ and $f := \top \sqsubseteq \bot$. However on the level of answering queries and introducing other possible answer-symbols having them as constants appears to be more natural.

Definition 3 ($\mathcal{ALC}$-satisfiability)**.** *Satisfiability* of formulae with respect to an interpretation $\mathcal{I} = (\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$ is defined inductively as follows:

- $\mathcal{I} \models t$ and not $\mathcal{I} \models f$
- $\mathcal{I} \models C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
- $\mathcal{I} \models \neg\psi$ iff not $\mathcal{I} \models \psi$ (abbreviated by $\mathcal{I} \not\models \psi$)
- $\mathcal{I} \models \varphi \wedge \psi$ iff $\mathcal{I} \models \varphi$ and $\mathcal{I} \models \psi$

A formula φ is valid iff for all interpretations $\mathcal{I}$ it holds $\mathcal{I} \models \varphi$.

Remark 4. It should be clear now that expressions of the form $C \sqsubseteq D$ and $\overline{C} \sqcap D$ are of two different types. The expression $C \sqsubseteq D$ is a formula, which thus can be true of false; whereas $\overline{C} \sqcap D$ is a concept, which is interpreted as a set of objects.

Definition 5 ($\mathcal{ALC}$-model)**.** Given a knowledge-base $\mathcal{KB}$, an interpretation $\mathcal{I}$ is a model of $\mathcal{KB}$, if every formula ψ in $\mathcal{KB}$ is satisfied ($\mathcal{I} \models \psi$). A knowledge-base is satisfiable if it has a model. A formula φ is a semantic consequence of $\mathcal{KB}$, written $\mathcal{KB} \models \varphi$, iff it is satisfied in every model of $\mathcal{KB}$. A knowledge-base $\mathcal{S}$ is a semantic consequence of $\mathcal{KB}$, written $\mathcal{KB} \models \mathcal{S}$, iff for all formulae $\psi \in \mathcal{S}$ we have $\mathcal{KB} \models \psi$.

Lemma 6 (Compactness)**.** *A knowledge-base $\mathcal{KB}$ is satisfiable iff every finite subset of $\mathcal{KB}$ is satisfiable.*

Proof. It is standard, see e.g. [4], to translate a concept C to a first-order formula with one free variable $f_C(x)$ such that for any interpretation $\mathcal{I} = (\Delta_{\mathcal{I}}, \cdot^{\mathcal{I}})$ we have

$$C^{\mathcal{I}} \text{ is exactly the set of all } x \text{ satisfying } f_C(x) \text{ under } \mathcal{I} \text{ as first-order model.}$$

Thus any $\mathcal{ALC}$-formula ψ can be translated to a first-order sentence f_ψ such that for any interpretation $\mathcal{I}$ we have

$$\mathcal{I} \text{ is an } \mathcal{ALC} \text{ model of } \psi \quad \text{iff} \quad \mathcal{I} \text{ is a first-order model of } f_\psi.$$

This translation is obtained by formalizing the notion of $\mathcal{ALC}$ satisfiability, i.e. Definition 3. The interesting case is concept inclusion: the first-order translation of $C \sqsubseteq D$ is given by

$$f_{C \sqsubseteq D} := \forall x (f_C(x) \to f_D(x))$$

where $f_C(x)$ and $f_D(x)$ are the first-order translations of the concepts C and D, respectively.
Now, compactness for $\mathcal{ALC}$ follows immediately from compactness of first-order logic. □

2.2.2 Incomplete Evaluation

In this paper we adapt an approach given by [10]. The setting allows to answer any Boolean queries on incomplete knowledge-bases. This means, that a formula can have one of three possible truth-values, namely *true* (t), *false* (f) or *unknown* (u), depending on whether the query, its negation or neither of both are semantic consequences of the underlying knowledge-base.

Definition 7. The evaluation function eval is defined as follows:

$$\mathsf{eval} : \begin{cases} \wp(\mathcal{L}_{\mathcal{ALC}}) \times \mathcal{L}_{\mathcal{ALC}} & \to & \{t, f, u\} \\ (\mathcal{KB}, \varphi) & \mapsto & \begin{cases} t & \text{if} & \mathcal{KB} \models \varphi \\ f & \text{if} & \mathcal{KB} \models \neg\varphi \text{ and } \mathcal{KB} \text{ is satisfiable} \\ u & \text{else} & \end{cases} \end{cases}$$

Remark 8 (Differences to propositional databases)**.** When we deal with incomplete databases over propositional logic it is sufficient to define the evaluator on the possible censored databases. Formally such a database is generated by taking a partition T, N, U of the propositional letters and constructing the observed database by $T \cup \{\neg n \mid n \in N\}$. So in a propositional (incomplete) database the database is somehow atomic.

Consider the following case: Let $T = \{a\}$, $N = \{b\}$ and $U = \{c\}$, then the formulae $c \to a$ and $b \to c$ would evaluate to t, and assuming that both formulae are not themselves protected, a reasonable censor should answer t as well. However, both a and $\neg b$ might be protected. So if the attacking agent next asks c, the only option left to the censor is to

refuse the answer. If the censor answers t it reveals a, with f it reveals $\neg b$ and worst with answer u both are revealed, by the definition of the evaluation function. Hence a censor in this setting might have to hide (non-trivial) more complex formulae than the ones to be protected in the first place. In [10] it is shown that for finite sets of protected formulae the disjunction of them has to be protected as well. Unfortunately this disjunction is likely to be a tautology, which leads to undesirable restrictions of the structure of the set of possible secrets.

We do not have this property in $\mathcal{ALC}$. Just adapting the atomicity approach would lead to the restriction that only sentences of form $C \sqsubseteq D$ and $\neg(C \sqsubseteq D)$ would be in the knowledge-base. However, more complex sentences like $A \sqsubseteq B \rightarrow C \sqsubseteq D$ could still be enforced by $\overline{A} \sqcup B \sqsubseteq \overline{C} \sqcup D$, which is a stronger property, but by far not as strong as $\neg(A \sqsubseteq B)$ or $C \sqsubseteq D$ would be. That means

$$\overline{A} \sqcup B \sqsubseteq \overline{C} \sqcup D \text{ entails } A \sqsubseteq B \rightarrow C \sqsubseteq D$$

but

$$\overline{A} \sqcup B \sqsubseteq \overline{C} \sqcup D \text{ neither entails } \neg(A \sqsubseteq B) \text{ nor } C \sqsubseteq D.$$

Also there is no natural way to justify a restriction to a certain level of formulae, like e.g. no use of conceptual operations within "axioms". This has the consequence that an answer unknown—as will be shown below—does not have the implicational strength that it possesses in the propositional case. Moreover within $\mathcal{ALC}$ from an answer unknown almost nothing can be inferred (this is established formally in Corollary 33). This leads to a more simple answer-selection strategy compared to the rather complex tables needed in [10].

As a consequence of this digression throughout this paper all knowledge-bases can contain arbitrary formulae and do especially not need to be satisfiable.

2.2.3 $\mathcal{CALC}$-Models

In addition to the presented evaluation function, we need a way to describe the attackers view on the knowledge-base created by the censors' answers. We will model the belief of an attacker after several questions by a set of $\mathcal{ALC}$-models, where each element represents a possible instance of the attackers (gained) knowledge.

Definition 9 (Cloud)**.** A ($\mathcal{CALC}$-) *cloud* is a pair $\mathfrak{C} = (W_{\mathfrak{C}}, \iota_{\mathfrak{C}})$, where

- $W_{\mathfrak{C}}$ is a nonempty set of worlds,
- for each $w \in W_{\mathfrak{C}}$, $\iota_{\mathfrak{C}}(w)$ is an $\mathcal{ALC}$-interpretation.

Definition 10 ($\mathcal{CALC}$-satisfiability)**.** Satisfiability of a formula $\Phi \in \mathcal{L}_{\mathcal{CALC}}$ within a $\mathcal{CALC}$-cloud $\mathfrak{C} = (W_{\mathfrak{C}}, \iota_{\mathfrak{C}})$ is given in the following way:

- $\mathfrak{C} \models \Box\psi$ iff for all $w \in W_{\mathfrak{C}}$ it is $\iota_{\mathfrak{C}}(w) \models \psi$
- $\mathfrak{C} \models \Diamond\psi$ iff there is a $w \in W_{\mathfrak{C}}$, s.t. $\iota_{\mathfrak{C}}(w) \models \psi$

A formula Φ is valid iff it is satisfied in all $\mathcal{CALC}$-clouds.

Definition 11 ($\mathcal{CALC}$-model)**.** A $\mathcal{CALC}$-cloud $\mathfrak{M}$ is a $\mathcal{CALC}$-model of a set $S \subseteq \mathcal{L}_{\mathcal{CALC}}$ if each formula $\Phi \in S$ is satisfied in $\mathfrak{M}$. A formula $\Phi \in \mathcal{L}_{\mathcal{CALC}}$ is semantically implied by S, written $S \models \Phi$, iff Φ is satisfied in all $\mathcal{CALC}$-models of S. Again we define for $T \subseteq \mathcal{L}_{\mathcal{CALC}}$

that T is semantically implied by S, written $S \models T$ iff for all $\Phi \in T$ it is $S \models \Phi$. For $T \subseteq \mathcal{L}_{\mathcal{CALC}}$ and $\Phi \in \mathcal{L}_{\mathcal{CALC}}$, we say that Φ is consistent with T if there exists a $\mathcal{CALC}$-model of $T \cup \{\Phi\}$.

Lemma 12 (Quartum non datur)**.** *Let $\psi \in \mathcal{L}_{\mathcal{ALC}}$ and let $\mathfrak{C}$ be a $\mathcal{CALC}$-cloud. Then exactly one of the following statements holds:*

- $\mathfrak{C} \models \{\Box\psi\}$
- $\mathfrak{C} \models \{\Box\neg\psi\}$
- $\mathfrak{C} \models \{\Diamond\psi, \Diamond\neg\psi\}$

Proof. Trivial. □

2.3 Privacy

When talking about privacy, we need to specify, not only what is to be kept secret, but also what means can be used to achieve it. We make use of three knowledge-bases, namely

- the (incomplete) *knowledge-base* $\mathcal{CK}$ (Censored Knowledge) concealed behind the censor,
- a set of *a priori-knowledge* $\mathcal{AK}$ (Attacker's Knowledge) describing the (incomplete and restricted) knowledge of the attacker (which, in this paper, is shared with the censor), and
- a set of *secrets* $\mathcal{SK}$ (Secret Knowledge) containing protected formulae.

Here we mean by protected that after any sequence of queries none of the formulae in $\mathcal{SK}$ may be revealed to the attacker. For the sake of simplicity we will assume that the attacker believes at the beginning only in true statements, i.e. we will assume $\mathcal{CK} \models \mathcal{AK}$.

Definition 13 (Privacy configuration)**.** A *privacy configuration* is a tripel

$$\mathcal{PC} = (\mathcal{CK}, \mathcal{AK}, \mathcal{SK}) \in \wp(\mathcal{L}_{\mathcal{ALC}}) \times \wp(\mathcal{L}_{\mathcal{ALC}}) \times \wp(\mathcal{L}_{\mathcal{ALC}})$$

such that

PC-A) $\mathcal{CK} \models \mathcal{AK}$ (Truthful start).

PC-B) $\mathcal{CK}$ is satisfiable (and hence so is $\mathcal{AK}$) (Consistency).

PC-C) $\mathcal{AK} \not\models \sigma$ for each $\sigma \in \mathcal{SK}$ (Hidden secrets).

Let us point out that $\mathcal{SK}$ does not need to be satisfiable. Moreover it even can contain formulae and their negations simultaneously.

Example 14 (Running example)**.** Consider the following setting:
A community of six persons (all with drivers licence) shares two cars, an Opol and a Persche. One day it happens that one of the cars was photographed in a speeding-trap. The photograph clearly shows the driver's hair colour and the car driven.
In order to determine who drove the car through the speed-trap the policeman calls at the community to inquire. The gardener (a very loyal employee) answers the phone.
In our terms we have the following situation: Both, $\mathcal{CK}$ (the knowledge of the gardener) and $\mathcal{AK}$ (the knowledge of the inquiring policeman), contain the following information:

- Alice, Bob, Carol, Dave, Eve and Floyd are Persons,

$$A \sqsubseteq \mathrm{Person} \sqcap B \sqsubseteq \mathrm{Person} \sqcap \ldots \sqcap F \sqsubseteq \mathrm{Person}$$

  Here A, B, C, D, E and F are quasi-nominals. A nominal is a concept that is satisfied by exactly one individual. In $\mathcal{ALC}$ we cannot express that a concept is a nominal but we can tacitly add information like $\neg(A \equiv \bot)$ or $(A \sqcap B) \equiv \bot$, which give us the desired properties.

- Opol and Persche are Cars and the car in question (TheCar) is one of them:

$$O \sqsubseteq \mathrm{Car} \sqcap P \sqsubseteq \mathrm{Car},\ \mathrm{TheCar} \equiv O \sqcup \mathrm{TheCar} \equiv P$$

  (again, O, P are quasi-nominals)

- Any Person is either blond, brunette or red-haired:

$$\mathrm{Red} \sqcup \mathrm{Blond} \sqcup \mathrm{Brunette} \equiv \mathrm{Person} \sqcap \mathrm{Red} \sqcap \mathrm{Blond} \equiv \bot \sqcap \ldots$$

- The community consists of exactly those persons:

$$\mathrm{Community} \equiv A \sqcup B \sqcup \ldots \sqcup F$$

- The car in question had only one driver, who is from the community:

$$\exists \mathrm{DriverOf}.\mathrm{TheCar} \equiv A \sqcup \ldots \sqcup \exists \mathrm{DriverOf}.\mathrm{TheCar} \equiv F$$

In addition, the policeman knows the hair colour ($\mathrm{HairColor}$) of the driver of the car ($\exists \mathrm{DriverOf}.\mathrm{TheCar}$), that is

$$\exists \mathrm{DriverOf}.\mathrm{TheCar} \sqsubseteq \mathrm{HairColor}$$

where $\mathrm{HairColor}$ is exactly one of Blond, Red or $\mathrm{Brunette}$. The policeman also knows the driven car (TheCar), which is either O or P. Hence we have

$$\mathrm{HairColor} \equiv \mathrm{Blond} \sqcap \mathrm{TheCar} \equiv O$$

or

$$\mathrm{HairColor} \equiv \mathrm{Red} \sqcap \mathrm{TheCar} \equiv P$$

or

$$\ldots.$$

Note that we have only one of them but not several simultaneously. We do not fix this knowledge now so that we can discuss several different settings.

To the knowledge of the gardener we add following:

- He knows the hair colours:

$$A, B, C \sqsubseteq \mathrm{Blond},\ D, E \sqsubseteq \mathrm{Brunette} \text{ and } F \sqsubseteq \mathrm{Red}$$

  (notice, that e.g. from this and the above information $\mathrm{Red} \sqcap \mathrm{Blond} \equiv \bot$ it follows $\neg F \sqsubseteq \mathrm{Blond}$, so the gardener knows the exact hair colour of community members)

- He has seen Alice, Carol and Floyd go to the carport and heard them leave by car:

$$\exists \mathrm{DriverOf}.\mathrm{TheCar} \sqsubseteq A \sqcup C \sqcup F$$

- If they took the Persche certainly Floyd was its driver:

$$\mathrm{TheCar} \equiv P \rightarrow (F \equiv \exists \mathrm{DriverOf}.P \wedge (A \sqcup C) \sqsupseteq \exists \mathrm{DriverOf}.O)$$

  (notice, that $\exists \mathrm{DriverOf}.O \sqsubseteq (A \sqcup C)$ does not mean they actually took the other car, since $\exists \mathrm{DriverOf}.O) \equiv \bot$ could hold.)

Since the gardener does not want one of the group to be fined, he must not give the policeman a chance to infer who drove that car.
Hence the secrets are

$$A \equiv \exists \mathrm{DriverOf}.\mathrm{TheCar},\ B \equiv \exists \mathrm{DriverOf}.\mathrm{TheCar},\ \ldots,\ F \equiv \exists \mathrm{DriverOf}.\mathrm{TheCar}$$

So far we do not have a privacy configuration, since $\mathcal{CK} \models \mathcal{AK}$ does not hold. However, once the policeman told (prior to start his inquiries) the gardener that the community's Persche was photographed by a speed-camera (i.e. $\mathrm{TheCar} \equiv P$), and hence the gardener knows

$$F \equiv \exists \mathrm{DriverOf}.\mathrm{TheCar}$$

this is achieved, since now also the driver's hair colour (red)

$$\exists \mathrm{DriverOf}.\mathrm{TheCar} \sqsubseteq \mathrm{Red}$$

can be inferred by the gardener.
In order to establish a privacy configuration in the situation where the Opol was driven, the policeman has to give out both information:

$$\mathrm{TheCar} \equiv O \quad \text{and} \quad \exists \mathrm{DriverOf}.\mathrm{TheCar} \sqsubseteq \mathrm{HairColor}$$

We define two query sequences of the policeman to provide example-answers of the presented censor-functions:

$$\begin{aligned}
\mathbf{P}^1 := (&\exists \mathrm{DriverOf}.\mathrm{TheCar} \equiv A, \exists \mathrm{DriverOf}.\mathrm{TheCar} \equiv B,\\
&\ldots, \exists \mathrm{DriverOf}.\mathrm{TheCar} \equiv F, t, t, \ldots)\\
\mathbf{P}^2 := (&\exists \mathrm{DriverOf}.\mathrm{TheCar} \sqsubseteq \mathrm{HairColor},\\
&A \sqsubseteq \overline{\mathrm{HairColor}}, B \sqsubseteq \overline{\mathrm{HairColor}}, \ldots, F \sqsubseteq \overline{\mathrm{HairColor}}, t, t, \ldots)
\end{aligned}$$

We keep these queries very simple in order to not increase the complexity of this already very long example set-up. The first sequence asks only the hidden secrets, the second only information on the hair-colours.

To achieve protection of the secret formulae, one might want to add the possibility of answering something else than (but perhaps "close to") the full truth. In particular, we want a mechanism that responds to a querying agent such that after any sequence of queries privacy is still maintained.
This can be implemented by a censor function.
We use $\mathbb{N} := \{1, 2, 3, \ldots\}$ to denote the positive and $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$ to denote all natural numbers, in order to simplify dealing with start-conditions.

Definition 15 (Censor)**.** A *censor* is a mapping that assigns an answering function

$$\mathrm{Cens}_{(\mathcal{OC},\mathcal{AC},\mathcal{SC})} : \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}} \to \mathbb{A}^{\mathbb{N}}$$

to each given privacy configuration. A sequence $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ is called a *query-sequence*. The set $\mathbb{A}$ contains the potential answers a censor might give. In this paper only $\{t, f, u, r\}$ and $\{t, f, u\}$ are possible choices for $\mathbb{A}$.

This definition turns out to provide simplicity in the more technical arguments. Especially the need for an additional logging data-structure vanishes. However, to guarantee non-usage of "future" queries and answers, we need a continuity-property, which we define below.
So far a censor can randomly answer and does not provide any safety.

Example 16 (Evaluation censors)**.** A trivial censor is the revealing evaluation censor that assigns the actual answer to each query:

$$\mathrm{Cens}_{(\mathcal{OC},\mathcal{AC},\mathcal{SC})}(\mathbf{q}) = (\mathsf{eval}(\mathcal{OC}, q_i))_{i\in\mathbb{N}}$$

A better, but also not very convenient censor is the overprotective evaluation censor given by

$$\mathrm{Cens}_{(\mathcal{OC},\mathcal{AC},\mathcal{SC})}(\mathbf{q}) = (\mathsf{eval}(\mathcal{AC}, q_i))_{i\in\mathbb{N}}$$

that tells the attacker only answers that it could calculate itself.

In order to qualify our gardener as an answering-function (here, the privacy configuration is fixed), he needs to be sure about the knowledge of the policeman. So we assume, he himself has some experience with photographs taken by speeding-cameras and hence knows, that only hair-colours and license-plates are visible on them. To upgrade him to a censor, we would have to make him independent of the observed situation as well. E.g. he would have to be able to react even if no one drove or the policeman had less or more knowledge (as long as all secrets are kept in the start) or even in a completely different start-situation (like no knowledge at all).
So equipped, our gardener can choose both of the above "strategies". However neither of these is a good choice. The revealing strategy is trivially no choice, since—so far our assumption—he wants to protect his employers, but would confirm that Floyd drove the car or imply this, e.g. by ruling out all others. So with the trivial censor our gardener would answer (for $\mathrm{TheCar} \equiv P$):

$$\mathrm{Cens}_{\ldots}(\mathbf{P}^1) = (f, f, f, f, f, t, t, t, \ldots)$$

$$\mathrm{Cens}_{\ldots}(\mathbf{P}^2) = (t, t, t, t, t, t, f, t, t, \ldots)$$

In both sequences the policeman has the perpetrator after the sixth answer.
With the overprotective approach on the other side, he might raise the policeman's suspicion, since the policeman might conclude (on a meta level) that the gardener must know all details to be able to copy his knowledge. For example, because the gardener "told him" (in our view confirmed) the red hair-colour of the driver (again with $\mathrm{TheCar} \equiv P$).

$$\mathrm{Cens}_{\ldots}(\mathbf{P}^1) = (u, u, u, u, u, u, t, t, \ldots)$$

$$\mathrm{Cens}_{\ldots}(\mathbf{P}^2) = (t, u, u, u, u, u, u, t, t, \ldots)$$

In order to formulate the conditions on whether a censor is considered "good", we need a translation of the content of a given answer.

Definition 17 (Content)**.** Let $\psi \in \mathcal{L}_{\mathcal{ALC}}$ and $a \in \mathbb{A} \subseteq \{t, f, u, r\}$. The *content* of a as response to ψ is given by

$$\mathrm{Cont}(\psi, a) = \begin{cases} \{\Box\psi\} & \text{if} \quad a = t \\ \{\Box\neg\psi\} & \text{if} \quad a = f \\ \{\Diamond\psi, \Diamond\neg\psi\} & \text{if} \quad a = u \\ \emptyset & \text{if} \quad a = r \end{cases}$$

Definition 18. Let $\mathcal{PC} = (\mathcal{OK}, \mathcal{AK}, \mathcal{SK})$ be a privacy configuration.
We define the *state cloud* wrt. a query-sequence $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ at stage n by

$$\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) := \bigcup_{\varphi \in \mathcal{AK}} \mathrm{Cont}(\varphi, t) \cup \bigcup_{i=1}^{n} \mathrm{Cont}(q_i, a_i),$$

where $\mathbf{a} := \mathrm{Cens}_{(\mathcal{OK},\mathcal{AK},\mathcal{SK})}(\mathbf{q})$. A censor Cens is called

- *credible* for $\mathcal{PC}$ iff for every sequence $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ and every $n \in \mathbb{N}$, it holds

$$\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \text{ is satisfiable} \qquad (C^n_{\mathcal{PC},\mathbf{q}})$$

- *effective* for $\mathcal{PC}$ iff for all sequences $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ and every $n \in \mathbb{N}$ it holds

$$\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \not\models \Box\sigma \text{ for every } \sigma \in \mathcal{SK} \qquad (E^n_{\mathcal{PC},\mathbf{q}})$$

  (i.e. no secret is semantically implied by a state cloud)

- *continuous* for $\mathcal{PC}$ iff for all sequences $\mathbf{q}, \mathbf{r} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ and all $n \in \mathbb{N}$, it is

$$\mathbf{q}|_n = \mathbf{r}|_n \rightarrow \mathrm{Cens}_{\mathcal{PC}}(\mathbf{q})|_n = \mathrm{Cens}_{\mathcal{PC}}(\mathbf{r})|_n \quad ,$$

  where $\mathbf{a}|_n$ denotes the initial segment of $\mathbf{a}$ of length n, i.e. $(a_1, \ldots, a_n)$.

A censor is called credible [effective, continuous], if it is credible [effective, continuous] for every privacy configuration. A censor is called credible [effective, continuous] up to stage n if the corresponding conditions hold for all state-clouds up to n.

Since from the context it will always be clear which censor is used to create the sets $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$, we omit the censor's name from the naming of the sequence. However, due to the high dependence on privacy configuration and query-sequence, we keep this information. Let us point out that the elements of the sets $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ heavily depend on all of these parameters.

Example 19. For our speeding setting as privacy configuration the gardener is credible if his answers are consistent with the knowledge of the policemen and he does not end up contradicting himself. He is effective, if at any stage of the telephone-call the policeman cannot infer, who drove that car. And he is continuous, since he is as gardener not skilled with the power of divination and hence cannot foresee the questions of the policeman.

The state cloud represents the view of the censor on the information given to the attacker by successive answering. In essence a censor is credible, if at any stage of answering the resulting state cloud is satisfiable (and therefore provides a "consistent" believe to the attacker), it is effective, if no secret is revealed by any initial segment of answers and it is continuous if the n-th answer does only depend on questions q_i and their answers a_i when $i < n$, and especially not on the full sequence.

Example 20. The revealing evaluation censor above is credible, but not effective. The over-protective evaluation censor is effective and credible. The censor given by

$$\mathrm{Cens}_{(\mathcal{OK},\mathcal{AK},\mathcal{SK})}(\mathbf{q}) = \begin{cases} (f)_{i\in\mathbb{N}} & \text{if} \quad \mathcal{SK} = \emptyset \\ (\mathsf{eval}(\mathcal{AK}, q_i))_{i\in\mathbb{N}} & \text{else} \end{cases}$$

is effective, but not credible. Effectiveness follows in the "else"-case by the definition of $\mathcal{PC}$, which implies $\mathsf{eval}(\mathcal{AK}, \sigma) \in \{f, u\}$ for all secrets $\sigma \in \mathcal{SK}$. If there are no secrets this fact is trivial.
However the censor is not credible, since it will answer f to a query on t in any privacy configuration with an empty set of secrets.

Effective but not credible censors are, however, not very common. The presented censor for example is credible for all privacy configurations that do contain at least one secret. Furthermore in the above construction one can mainly change the answering function in case the set of secrets is empty and change to a different effective and credible censor in the case when there is something to be kept secret. This is due to the fact that if the censors' answers lead to an unsatisfiable state cloud at stage n, for any secret σ (in fact for any formula $\sigma \in \mathcal{L}_{\mathcal{ALC}}$) it would follow $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) \models \Box\sigma$ immediately, violating the property of effectiveness. To summarize this:

Lemma 21. *Let $\mathcal{PC}$ be a privacy configuration, s.t. $\mathcal{SK} \neq \emptyset$. Then every censor that is effective for $\mathcal{PC}$ is also credible for $\mathcal{PC}$.*

Lemma 22. *A censor is*

- *credible for $\mathcal{PC}$ iff $\mathscr{HC}_{\mathcal{PC},\mathbf{q}} := \bigcup_{i=0}^{\infty} \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(i)$ is satisfiable*
- *effective for $\mathcal{PC}$ iff for no $\sigma \in \mathcal{SK}$ it holds $\mathscr{HC}_{\mathcal{PC},\mathbf{q}} \models \Box\sigma$*

Proof. Application of compactness of $\mathcal{ALC}$:
Ad "credible": The direction right to left is trivial.
From left to right:
Observe that by definition every formula $\Phi \in \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(i)$ starts with a modality and contains exactly this modality.
Let $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ fixed.
Define $K_\Box := \{\Box\varphi \mid \Box\varphi \in \mathscr{HC}_{\mathcal{PC},\mathbf{q}}\}$ and $K_{\Diamond\varphi} := K_\Box \cup \{\Diamond\varphi\}$ for each $\Diamond\varphi \in \mathscr{HC}_{\mathcal{PC},\mathbf{q}}$.

Let $K \subseteq K_\Box$ be finite. Then K is subset of some $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(i)$. So it is satisfiable by premise. Hence there exists a $\mathcal{CALC}$-model with a world w such that $\iota(w) \models \{\varphi \mid \Box\varphi \in K\}$ (Here, of course, $\models$ refers to $\mathcal{ALC}$-satisfication). Since K was arbitrary, by compactness of $\mathcal{ALC}$ there is a $\mathcal{ALC}$-interpretation $\mathcal{W}_\Box$ with $\mathcal{W}_\Box \models \{\psi \mid \Box\psi \in K_\Box\}$.
Analogous we find $\mathcal{ALC}$-interpretations $\mathcal{W}_{\Diamond\varphi}$ s.t.

$$\mathcal{W}_{\Diamond\varphi} \models \{\psi \mid \Box\psi \in K_{\Diamond\varphi} \text{ or } \Diamond\psi \in K_{\Diamond\varphi}\}.$$

Define the clouded-model of $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}$ by setting

- $W := \{\mathcal{W}_\Box\} \cup \bigcup_{\Diamond\varphi \in \mathscr{HC}_{\mathcal{PC},\mathbf{q}}} \{\mathcal{W}_{\Diamond\varphi}\}$,
- $\iota(\mathcal{W}) := \mathcal{W}$

This model satisfies $\mathscr{H}_{\mathcal{PC},\mathbf{q}}$.
Ad "effective": analogous. □

Example 23. By this theorem it follows that our gardener is credible if his answers are consistent after the (infinitely long) call has ended. He is effective if at this stage the policeman does not have any idea about who drove the Persche.

By our definition of effectiveness the secret knowledge is defined positive in the sense that it can only protect the truth of a Boolean statement, the statement itself might however be a negation. In this setting it is not possible to protect an "unknown"-response. This will be exploited to treat censors that cannot refuse but may lie.

3 Censors

As mentioned above continuity is one of the presumptions needed to effectively calculate an answer to the question next in sequence.
To achieve this in [10] a logging facility is introduced. However, since the censors here have access to the full sequence of questions, the equivalent to the log-file after answering question n is just $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ as defined.

Lemma 24. *Let* Cens *be a credible and effective censor,* $n \in \mathbb{N}$ *and* $\mathcal{KB} := \{\psi \mid \Box\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)\}$, *then the following hold:*

a) $\mathcal{KB}$ *is satisfiable.*

b) $\mathsf{eval}(\mathcal{KB}, \psi) \in \{u, f\}$ *for each* $\psi \in \mathcal{SK}$

c) $\mathsf{eval}(\mathcal{KB}, \psi) = t$ *if* $\Box\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$

d) $\mathsf{eval}(\mathcal{KB}, \psi) = u$ *if* $\Diamond\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ *(or if* $\Diamond\neg\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$*)*

Proof. Ad a): Since Cens is credible there is a clouded-model (W, ι) of $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$. For $w \in W$ by definition $\iota(w)$ satisfies $\mathcal{KB}$.
Ad b): Since $\{\Box\psi \mid \psi \in \mathcal{KB}\} \subseteq \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ this follows by definition of effectiveness.
Ad c): By definition of $\mathcal{ALC}$-satisfiability ψ must be semantically implied by $\mathcal{KB}$, hence by definition of eval the statement follows.
Ad d): By construction of $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ from Cont whenever $\Diamond\psi$ or $\Diamond\neg\psi$ is contained in $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$, the other one is included as well. Hence by credibility, in the cloud-model (W, ι) of $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$, there are worlds $w_1, w_2 \in W$, such that $\iota(w_1) \models \psi$ and $\iota(w_2) \models \neg\psi$. As in a) $\iota(w_1)$ and $\iota(w_2)$ are models of $\mathcal{KB}$. Hence neither ψ nor $\neg\psi$ can be valid in $\mathcal{KB}$. By definition of eval follows the proposition. □

Some properties one might deem useful or desirable for a censor. Those include being *truthful*, namely not making the querying agent believe in something false, *minimal invasive*, meaning to only hide answers, that are *directly* and *truly* harmful (i.e. revealing protected information), *cooperative*, satisfying the wish for information of the querying agent, and *repudiating*, which provides safety on a level of meta inference. Since being or seeming cooperative basically translates to either giving up on hiding information, and hence the intent of censoring, or to lie and not refuse whenever necessary, we omit a formal definition. Although *seemingly* cooperative censors are treated below in section 3.2. To the three other terms we will provide formal definitions and discussions on their features.

Definition 25 (Truthful)**.** The censor Cens is called *truthful* iff for all privacy configurations, for all question sequences $\mathbf{q}$ and for all i:

$$a_i \in \{r, \mathsf{eval}(\mathcal{CK}, q_i)\} \ ,$$

where $\mathbf{a} := \mathrm{Cens}(\mathbf{q})$.
A censor that is not truthful is called *lying*.

A truthful censor must not lie, but can refuse to answer in order to protect sensitive information. So the above definition means that the censor either provides the correct answer to a query or refuses to answer, hence the name truthful.
There is an slightly less intuitive characterisation of truthful censors via the following translation, which we use to show that every truthful censor is credible:

Definition 26 (Cloud-translation)**.** Let $\mathcal{KB} \subseteq \mathcal{L}_{\mathcal{ALC}}$ be a knowledge-base. Then the set

$$\mathrm{ClTr}(\mathcal{KB}) := \bigcup_{\psi \in \mathcal{L}_{\mathcal{ALC}}} \mathrm{Cont}(\psi, \mathsf{eval}(\mathcal{KB}, \psi))$$

is called (universal) *cloud translation* of $\mathcal{KB}$.

Some facts are immediate:

Proposition 27 (Properties)**.** *Let $\mathcal{KB}$ be an arbitrary knowledge-base, let $\psi \in \mathcal{L}_{\mathcal{ALC}}$ and let $\mathfrak{C}$ be an $\mathcal{ALC}$-cloud. The following statements hold*

- *If $\mathfrak{C} \models \mathrm{ClTr}(\mathcal{KB})$, then $\mathfrak{C} \models \Box\psi$ iff $\Box\psi \in \mathrm{ClTr}(\mathcal{KB})$*
- *If $\mathrm{ClTr}(\mathcal{KB}) \models \{\Diamond\psi, \Diamond\neg\psi\}$ and $\mathcal{KB}$ is satisfiable, then $\psi, \neg\psi \notin \mathcal{KB}$.*
- *At least one of the formulae $\Box\psi$, $\Box\neg\psi$ or $\Diamond\psi$ is an element of $\mathrm{ClTr}(\mathcal{KB})$.*
- $\mathrm{Cont}(\psi, \mathsf{eval}(\mathcal{KB}, \psi)) \subseteq \mathrm{ClTr}(\mathcal{KB})$.
- *Let $\mathcal{V} := \{\eta \in \mathcal{L}_{\mathcal{ALC}} \mid \Box\eta \in \mathrm{ClTr}(\mathcal{KB})\}$.*
  Then $\mathsf{eval}(\mathcal{KB}, \psi) = \mathsf{eval}(\mathcal{V}, \psi)$ and $\mathcal{V} = \{\psi \in \mathcal{L}_{\mathcal{ALC}} \mid \mathcal{KB} \models \psi\}$

Lemma 28 (Cloud-translation preserves satisfiability)**.**
Let $\mathcal{KB}$ be a knowledge-base. Then $\mathcal{KB}$ is $\mathcal{ALC}$-satisfiable iff $\mathrm{ClTr}(\mathcal{KB})$ is $\mathcal{CALC}$-satisfiable

Proof. Left to right:
Let $U := \{\psi \in \mathcal{L}_{\mathcal{ALC}} \mid u = \mathsf{eval}(\mathcal{KB}, \psi)\}$.
Assume $U \neq \emptyset$.
By definition of the evaluation for all $\psi \in U$ there are interpretations $\mathcal{I}_\psi$ and $\mathcal{J}_\psi$, such that $\mathcal{I}_\psi \models \mathcal{KB} \cup \{\psi\}$ and $\mathcal{J}_\psi \models \mathcal{KB} \cup \{\neg\psi\}$.
Define $\mathfrak{C}$ by $W_\mathfrak{C} := U \times \{t, f\}$ and $\iota_\mathfrak{C}$ by $\iota_\mathfrak{C}((\psi, t)) = \mathcal{I}_\psi$ and $\iota_\mathfrak{C}((\psi, f)) = \mathcal{J}_\psi$. Hence by choice of $\mathcal{I}_\psi$ and $\mathcal{J}_\psi$ the following are immediate:

- $\mathfrak{C} \models \Box\varphi$ for all φ with $\mathsf{eval}(\mathcal{KB}, \varphi) = t$,
- $\mathfrak{C} \models \Diamond\psi$ for all ψ with $\mathsf{eval}(\mathcal{KB}, \psi) = u$ ($\psi \in U$) and
- $\mathfrak{C} \models \Diamond\neg\psi$ for all ψ with $\mathsf{eval}(\mathcal{KB}, \psi) = u$ ($\psi \in U$).

For the formulae φ with $\mathsf{eval}(\mathcal{KB}, \varphi) = f$ we have by definition $\mathcal{KB} \models \neg\varphi$. Hence all $\mathcal{I}_\psi \not\models \varphi$ and $\mathcal{J}_\psi \not\models \varphi$ and hence $\mathfrak{C} \models \Box\neg\varphi$ for all such formulae. Therefore $\mathfrak{C} \models \mathrm{ClTr}(\mathcal{KB})$.
If $U = \emptyset$ ($\mathcal{KB}$ is complete), let $\mathcal{I} \models \mathcal{KB}$ be assumption. Then we have that $\mathfrak{C}$ with $W_\mathfrak{C} = \{w\}$ and $\iota_\mathfrak{C}(w) := \mathcal{I}$ is a model of $\mathrm{ClTr}(\mathcal{KB})$ as is easily seen.
Right to left:
Let $\mathfrak{C}$ be a model of $\mathrm{ClTr}(\mathcal{KB})$. By definition $\mathfrak{C} \models \{\Box\psi \mid t = \mathsf{eval}(\mathcal{KB}, \psi)\}$. Hence for any $w \in W_\mathfrak{C}$ it is $\iota_\mathfrak{C}(w) \models \mathcal{KB}$. □

Lemma 29 (Truth by cloud-translation)**.** *A censor* Cens *is truthful iff for every privacy configuration $\mathcal{PC} = (\mathcal{CK}, \mathcal{AK}, \mathcal{SK})$, every query sequence $\mathbf{q}$ and every $n \in \mathbb{N}_0$ we have*

$$\mathrm{ClTr}(\mathcal{CK}) \models \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n).$$

Proof. Left to right:
We show $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) \subseteq \mathrm{ClTr}(\mathcal{CK})$ by induction on n:
Since $\mathcal{CK} \models \mathcal{AK}$, then, for every $\psi \in \mathcal{AK}$, we have that $\mathsf{eval}(\mathcal{CK}, \psi) = t$.
Hence

$$\begin{aligned}\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(0) &= \{\Box\psi \mid \psi \in \mathcal{AK}\} \\ &= \bigcup_{\psi\in\mathcal{AK}} \mathrm{Cont}(\psi, \mathsf{eval}(\mathcal{KB}, \psi)) \subseteq \mathrm{ClTr}(\mathcal{CK})\end{aligned}$$

Step: Since Cens is truthful, $a_{n+1} \in \{r, \mathsf{eval}(\mathcal{CK}, q_{n+1})\}$. Thus either

$$\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n+1) = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) \cup \mathrm{Cont}(q_{n+1}, r) = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n)$$

and we are done by I.H. or

$$\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n+1) = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) \cup \mathrm{Cont}(q_{n+1}, \mathsf{eval}(\mathcal{CK}, q_{n+1}))$$

which follows by I.H. and $\mathrm{Cont}(q_{n+1}, \mathsf{eval}(\mathcal{CK}, q_{n+1})) \subseteq \mathrm{ClTr}(\mathcal{CK})$ by definition of ClTr.

Right to left:
Assume there is an index n, s.t. $a_n \notin \{r, \mathsf{eval}(\mathcal{CK}, q_n)\}$. Wlog. let this index be minimal for $\mathbf{q}$. Let $\mathfrak{C}$ be a cloud model, s.t. $\mathfrak{C} \models \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n)$. Then $\mathfrak{C} \not\models \mathrm{Cont}(q_n, \mathsf{eval}(\mathcal{CK}, q_n))$ by Lemma 12. Hence (in fact) no model $\mathfrak{C}$ of $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n)$ satisfies $\mathfrak{C} \models \mathrm{ClTr}(\mathcal{CK})$. But By Lemma 28 there is at least one model of $\mathrm{ClTr}(\mathcal{CK})$, since $\mathcal{CK}$ is satisfiable by definition of privacy configuration. We conclude that this model of $\mathrm{ClTr}(\mathcal{CK})$ cannot be a model of $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n)$ and, therefore, $\mathrm{ClTr}(\mathcal{CK}) \not\models \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n)$ as required. □

The previous two lemmata combine very nicely:

Corollary 30. *Every truthful censor is credible.*

A good censor should only fail to deliver accurate information, whenever a truthful answer would lead to a revealed secret. Formally this translates to the following property:

Definition 31 (Minimally invasive)**.** Let Cens be effective and credible. Cens is called *minimally invasive* iff whenever $a_i \neq \mathsf{eval}(\mathcal{CK}, q_i)$ replacing a_i by $\mathsf{eval}(\mathcal{CK}, q_i)$ would lead to a violation of either effectiveness or credibility.

In essence a truthful answer should be hidden only if this is really necessary.

The two definitions of minimally invasive and truthful are not only reasonable in a philosophical view on censors. Also various trivial censors like "only refusing" can be disregarded. In summary there remain three acceptable censor-variants: The *truthful* (but refusing) censors, the *(cooperative,) minimally invasive and lying* (and not refusing) censors, and the *minimally invasive combined* censors (lying and refusing).

One remarkable feature of the expressive strength of a non-atomic incomplete database raises from the two facts we discuss next. Namely that, to some extent, answering u (meaning "I don't know") to a query cannot breach effectiveness or credibility. However, this method only works as long as hiding ignorance (i.e. hiding that "I don't know" actually is the case) is not allowed.

Lemma 32. *Let $\varphi, \eta \in \mathcal{L}_{\mathcal{ALC}}$ and let* Cens *be a censor. Further assume that each of $\Diamond\varphi$, $\Diamond\neg\varphi$, and $\Diamond\eta$ is consistent with $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$. Then if*

$$\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \cup \mathrm{Cont}(\varphi, u) \models \Box\eta \ ,$$

it follows $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \models \Box\eta$.

Proof. Since $\Diamond\varphi, \Diamond\neg\varphi$ are satisfiable in $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$, there are cloud-models

$$\mathfrak{L} \models \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \cup \{\Diamond\varphi\} \quad \text{and} \quad \mathfrak{M} \models \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \cup \{\Diamond\neg\varphi\}.$$

Hence there are worlds $l \in W_{\mathfrak{L}}$ and $m \in W_{\mathfrak{M}}$ with $\mathcal{I} := \iota_{\mathfrak{L}}(l) \models \varphi$, $\mathcal{J} := \iota_{\mathfrak{M}}(m) \models \neg\varphi$ and for all formulae $\rho \in \mathcal{L}_{\mathcal{ALC}}$, s.t. $\Box\rho \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$, $\mathcal{I} \models \rho$ and $\mathcal{J} \models \rho$.

Let $\mathfrak{C}$ be an arbitrary cloud-model of $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ and $w \in W_{\mathfrak{C}}$. Then in $\iota_{\mathfrak{C}}(w)$ it holds either φ or $\neg\varphi$. Assume φ holds:

By adding a fresh world j to $W_{\mathfrak{C}}$ with $\iota_{\mathfrak{C}}(j) = \mathcal{J}$ we obtain a new model that satisfies $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \cup \mathrm{Cont}(\varphi, u)$, since by construction all $\Box$-formulae are satisfied and for each $\Diamond$-formula there is at least one world satisfying the corresponding $\mathcal{ALC}$-formula. Let us point out, that this is sufficient only because no further logical connectives appear in $\mathcal{CALC}$, especially no kind of disjunction or negation.

Thus by presumption this model satisfies $\Box\eta$. Therefore by definition η is satisfied in all $\iota(w)$ where $w \in W_{\mathfrak{C}} \cup \{j\}$. Hence η is satisfied in all $\iota(w)$ where $w \in W_{\mathfrak{C}}$ and hence $\mathfrak{C} \models \Box\eta$.

The case $\neg\varphi$ follows analogously by adding $\mathcal{I}$. □

Corollary 33 (Security in ignorance)**.** *Let* Cens *be a censor. For privacy configuration $\mathcal{PC}$, query-sequence $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$ and $\mathbf{a} := \mathrm{Cens}_{\mathcal{PC}}(\mathbf{q})$, let* Cens *fulfil the conditions $(C^n_{\mathcal{PC},\mathbf{q}})$ and $(E^n_{\mathcal{PC},\mathbf{q}})$. If both $\Diamond q_{n+1}$ and $\Diamond\neg q_{n+1}$ are satisfiable in $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ then setting the corresponding answer to $a_{n+1} := u$ leads to satisfaction of the conditions $(C^{n+1}_{\mathcal{PC},\mathbf{q}})$ and $(E^{n+1}_{\mathcal{PC},\mathbf{q}})$.*

Remark 34 (Back adaptation to incomplete propositional logic)**.** If one is allowing full formulae not only in the databases of the attacker, but also in the censored database and in the set of secrets, then the presented methods work fine in the propositional case as well.

For example, observe the propositional setting, where $a \to s$ and $\neg a \to t$ are both answered to be true and where s, t are secrets. Querying a leaves the attacker on any possible non refused answer with a secret: if a is true s, if it is false t and both if it is unknown.

This possibility vanishes instantly by allowing all formulae in the database, since now it only follows that the sentences $a \to s$ and $\neg a \to t$ are in the (suspected) database in case "unknown" is answered. Also it follows that some of a, s, t and their negations are not in

the assumed to be protected database.
However looking at this simple setting, which could be generalized, we also see, that in case a, s, t are $\mathcal{ALC}$-formulae there are even more possibilities for the content of the censored knowledge-base. One was presented above in Remark 8.

In view of the fact that no algorithm can be hidden forever, an additional goal is to ensure that a continuous censor should provide unrevealing answers even if the method of determination is revealed and the attacker even knows the potential secrets.

Definition 35 (Repudiation)**.** A censor Cens is called *repudiating* if for each privacy configuration $(\mathcal{CK}, \mathcal{AK}, \mathcal{SK})$ and each query sequence $\mathbf{q}$ there are knowledge-bases $\mathcal{KB}_i$, s.t.

R-A) $\mathrm{Cens}_{\mathcal{CK},\mathcal{AK},\mathcal{SK}}(\mathbf{q})|_n = \mathrm{Cens}_{\mathcal{KB}_n,\mathcal{AK},\mathcal{SK}}(\mathbf{q})|_n$,

R-B) for all $i \in \mathbb{N}$ and all $\sigma \in \mathcal{SK} : \mathcal{KB}_i \not\models \sigma$,

R-C) for all i $(\mathcal{KB}_i, \mathcal{AK}, \mathcal{SK})$ is a privacy configuration.

The condition of repudiation intuitively reads that there is a knowledge-base in which all secrets are (simultaneously) not true and supplied to a censor would produce the same answers as the original. Notice that this definition provides a version of plausible deniability to all secrets, depending on the query sequence.

Example 36. As already mentioned above, the gardener concludes from the information that the Persche was driven, that the driver was redhead. However a policeman, who might be aware of the gardeners record of speeding-tickets might get suspicious about how the gardeners avoids answers to e.g. hair-colours. So -at least- he can infer on a meta-level that the gardener can directly infer the hair-colour of the driver. In the presented Persche-situation above, Repudiation is still not violated by the overprotective gardener, since the policeman can assume, that the gardener really does not know the persons' haircolors. A given interpretation of the created statecloud could *satisfy*, e.g. $\Box A, B \sqsubseteq \mathrm{Red}$, and another $\Box A, D, F \sqsubseteq \mathrm{Red}$.

A good candidate as such a cover-up-sequence of knowledge-bases turns out to be

$$\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) := \{\psi \mid \Box\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)\}$$

at least for effective censors. The main reason is the following fact:

Proposition 37. *For all $n \in \mathbb{N}$ if a censor is effective up to stage n it holds*

$$\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \models \Box\psi \text{ iff } \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \models \psi.$$

Proof. Right to left is trivial.
By effectiveness it exists a model of $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$.
Assume $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \not\models \psi$.
Let $\mathfrak{M} \models \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ and $\mathcal{I} \models \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ with $\mathcal{I} \not\models \psi$.
Hence $\mathcal{I} \models \neg\psi$.
Then the model constructed by $\mathfrak{N} = (W_{\mathfrak{N}}, \iota_{\mathfrak{N}})$ with $W_{\mathfrak{N}} := W_{\mathfrak{M}} \cup \{i\}$ and

$$\iota_{\mathfrak{N}}(w) = \begin{cases} \iota_{\mathfrak{M}}(w) & \text{if} \quad w \in W_{\mathfrak{M}} \\ \mathcal{I} & \text{if} \quad w = i \end{cases}$$

is a model of $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$. But $\mathfrak{N} \not\models \Box\psi$. □

Corollary 38. *Let* Cens *be truthful. Then* $\mathcal{CK} \models \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ *for any privacy configuration* $\mathcal{PC}$ *and all* n*.*

Proof. Let $\psi \in \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$.
By Prop. 37 $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \models \Box\psi$. Hence by Lemma 29 it is $\mathrm{ClTr}(\mathcal{CK}) \models \Box\psi$. By definition of the cloud translation it follows $\Box\psi \in \mathrm{ClTr}(\mathcal{CK})$ and by the same definition $\mathcal{CK} \models \psi$. □

At a first glance the condition to be repudiating might seem to be a direct consequence of effectiveness and continuity. However, it turns out that it is an additional property. Below we present the censor TCens, which is effective but not repudiating.

In the following sections we discuss each of the different kinds of censors and provide effective, credible and continuous censors that are repudiating whenever possible.

3.1 Truthful Censors

In this section the censors must be truthful. So they might refuse to answer, but they cannot lie. An interesting point in this setting is the possibility to complete the separation of effectiveness from repudiation. To this end we will discuss two truthful censors which are both continuous, effective and credible, but only one is repudiating. This will also show how a leak of the censor algorithm can present a way of obtaining secrets.

Algorithm 1 Calculate $\mathrm{RTCens}_{\mathcal{PC}}(\mathbf{q})$

Require: $\mathcal{PC} = (\mathcal{CK}, \mathcal{SK}, \mathcal{AK})$ as privacy configuration
Require: $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$
1: $\mathbf{a} = (a_1, a_2, \ldots) \leftarrow (u, u, \ldots)$
2: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(0) \leftarrow \bigcup_{\varphi \in \mathcal{AK}} \mathrm{Cont}(\varphi, t)$
3: **for** $n \leftarrow 1 \ldots \infty$ **do**
4: compliant $\leftarrow$ **true**
5: **for** $\sigma \in \mathcal{SK}$ **do**
6: **if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t) \models \Box\sigma$ **or** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, f) \models \Box\sigma$ **then**
7: $a_n \leftarrow r$
8: compliant $\leftarrow$ **false**
9: **end if**
10: **end for**
11: **if** compliant **then**
12: $a_n \leftarrow \mathsf{eval}(\mathcal{CK}, q_n)$
13: **end if**
14: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \leftarrow \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, a_n)$
15: **end for**
16: **return** $\mathbf{a}$

Definition 39 (Truthful censor-functions)**.** We denote the censor determined by Algorithm 1 as RTCens (repudiating, not minimally invasive truthful censor) and the censor determined by Algorithm 2 as TCens (non repudiating, minimally invasive truthful censor).

Algorithm 2 Calculate $\mathrm{TCens}_{\mathcal{PC}}(\mathbf{q})$

Require: $\mathcal{PC} = (\mathcal{CK}, \mathcal{SK}, \mathcal{AK})$ as privacy configuration
Require: $\mathbf{q} \in \mathcal{L}_{\mathcal{ALC}}^{\mathbb{N}}$
1: $\mathbf{a} = (a_1, a_2, \ldots) \leftarrow (u, u, \ldots)$
2: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(0) \leftarrow \bigcup_{\varphi \in \mathcal{AK}} \mathrm{Cont}(\varphi, t)$
3: **for** $n \leftarrow 1 \ldots \infty$ **do**
4: compliant $\leftarrow$ **true**
5: $p \leftarrow \mathsf{eval}(\mathcal{CK}, q_n)$
6: **for** $\sigma \in \mathcal{SK}$ **do**
7: **if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, p) \models \Box\sigma$ **then**
8: $a_n \leftarrow r$
9: compliant $\leftarrow$ **false**
10: **end if**
11: **end for**
12: **if** compliant **then**
13: $a_n \leftarrow p$
14: **end if**
15: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \leftarrow \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, a_n)$
16: **end for**
17: **return** $\mathbf{a}$

So the difference between both algorithms is when they refuse to answer. The second only refuses if a truthful answer leads to an $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(\cdot)$ in which a secret is valid. The first also refuses when a response of t or f would lead to this violation of effectiveness. It is immediately clear that RTCens is not minimally invasive.

At a first glance and having Corollary 33 in mind it appears, that RTCens should also answer unknown, if that is the evaluated answer. However this would lead to a censor violating repudiation.

Example 40 (Non-repudiation in truthful ignorance)**.** Assume RTCens would answer u, whenever $\mathsf{eval}(\mathcal{CK}, q_i) = u$. In this case the proofs of continuity, truth, credibility and effectiveness given in the coming lemmata still work fine (after shifting around some cases). We give a counter example to show a failure in repudiation:
Assume $\mathcal{PC} := (\mathcal{CK}, \mathcal{AK}, \mathcal{SK})$ with $\mathcal{CK} := \{\sigma\}$, $\mathcal{AK} := \emptyset$ and $\mathcal{SK} := \{\sigma, \rho\}$, with $\sigma, \rho \in \mathcal{L}_{\mathcal{ALC}}$ (Remark: This example works fine in the propositional case, too.).
We ask query sequence $\mathbf{q} := (\sigma \vee \rho, \rho, \sigma, t, t, \ldots)$.
As is easily calculated, we get:

$$\begin{aligned} \mathsf{eval}(\mathcal{CK}, \sigma \vee \rho) &= t \\ \mathsf{eval}(\mathcal{CK}, \rho) &= u \\ \mathsf{eval}(\mathcal{CK}, \sigma) &= t \end{aligned}$$

It is simple to infer the answer given by the modified $\mathrm{RTCens}_{\mathcal{PC}}$ (with unknown):

$$\mathbf{a} = (t, u, r, t, \ldots)$$

The violation of repudiation happens after the refusal:
First notice that after the second answer, any knowledge-base $\mathcal{KB}$ that produces the same

answers has to semantically imply $\sigma \vee \rho$, but must imply neither ρ nor $\neg\rho$. Hence there are two options left for σ: either it evaluates to u (meaning $\sigma \vee \rho$ is a consequence of more complex axioms) or it evaluates to t. Since in the first case our modified censor would answer u, which it does not ($a_3 = r$), there is only one option left and this is $\mathsf{eval}(\mathcal{KB}, \sigma) = t$.

Example 41 (14 cont'd)**.** Let us calculate the answers of both truthful censors in the case where $\mathrm{TheCar} \equiv P$:

$$\mathrm{TCens}\ldots(\mathbf{P}^1) = (f, f, f, f, r, r, t, t, \ldots)$$

$$\mathrm{TCens}\ldots(\mathbf{P}^2) = (t, t, t, t, t, r, r, t, t, \ldots)$$

The non-repudiating censor refuses to answer on two questions in both sequences. In $\mathbf{P}^1$, since correctly answering f to $\exists\mathrm{DriverOf}.P \equiv D$, would already imply $\exists\mathrm{DriverOf}.P \equiv F$ to be true in any interpretation.
Similarly in the answer to $\mathbf{P}^2$.
However, in contrast to Example 40 above, Floyd is still not lost when the policeman knows the algorithm, since it is clear, that f would be a safe answer to $\exists\mathrm{DriverOf}.P \equiv E$, as well as $\exists\mathrm{DriverOf}.P \equiv F$. So both cases remain as possible interpretations.
For the repudiating version, we obtain the answers:

$$\mathrm{RTCens}\ldots(\mathbf{P}^1) = (r, r, r, r, r, r, t, t, \ldots)$$

$$\mathrm{RTCens}\ldots(\mathbf{P}^2) = (t, t, t, t, t, r, r, t, t, \ldots)$$

In the first answer, since every answer to true would immediately yield a secret. And in the second query's answer, which is the same answer that TCens gave to $\mathbf{P}^2$, by understanding that changing any of the given r to either f or t would give away one of the community-members as driver. However, the given answer rules out A, B, C and D as possible drivers.

The continuity of both censors is immediate:

Lemma 42 (Continuity)**.** *The censors* RTCens *and* TCens *are continuous.*

Proof. Clear by inspection of the algorithm: All decisions are based only on the state-clouds that are constructed in a step before and the current query. □

Lemma 43 (Truth)**.** *The censors* RTCens *and* TCens *are truthful.*

Proof. In both algorithms the answer is only modified (if at all) to r. Hence the condition $a_n \in \{r, \mathsf{eval}(\mathcal{CK}, q_n)\}$ is always satisfied. □

The previous lemma in combination with Corollary 30 provides immediately:

Corollary 44 (Credibility)**.** *The censors* RTCens *and* TCens *are credible.*

Lemma 45 (Effectiveness)**.** *The censors* RTCens *and* TCens *are effective.*

Proof. Let $\mathrm{Cens} \in \{\mathrm{RTCens}, \mathrm{TCens}\}$, $\mathcal{PC}$ be a privacy configuration and $\mathbf{q}$ be a query sequence. Set $\mathbf{a} := \mathrm{Cens}_{\mathcal{PC}}(\mathbf{q})$ and assume that for all $m < n$ the required property - for all $\sigma \in \mathcal{SK}$ not $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(m) \models \Box\sigma$ - holds. We prove that for n this holds as well:
Case $\mathsf{eval}(\mathcal{CK}, q_n) = u$:
For both TCens and RTCens (also in the modification of Example 40):
In case u is selected as answer, effectiveness in stage n is immediate from Corollary 33. Only RTCens can also refuse in this case. Then it is

$$\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \emptyset = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n-1)$$

and the property follows by induction hypothesis.
Case $\mathsf{eval}(\mathcal{CK}, q_n) = t$ (for both censors):
If the Property is violated, there is a $\sigma \in \mathcal{SK}$, s.t. $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) \models \Box\sigma$. But, by construction of the state-cloud, we have

$$\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t)$$

and hence $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t) \models \Box\sigma$ in contradiction to the refusal-selection in line 7 in TCens and line 6 in RTCens, respectively. Hence both censors would have refused to answer then leaving

$$\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \emptyset$$

and thus fulfilling the property by I.H.
Case $\mathsf{eval}(\mathcal{CK}, q_n) = f$ (for both censors): follows analogously. □

Lemma 46 (Repudiation)**.** *The censor* RTCens *is repudiating.*

Proof. Let $\mathcal{PC} = (\mathcal{CK}, \mathcal{AK}, \mathcal{SK})$ be a privacy configuration and $\mathbf{q}$ be a query sequence. Set $\mathbf{a} := \mathrm{RTCens}_{\mathcal{PC}}(\mathbf{q})$.
We show that $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ is a possible choice.
Ad R-C): $\mathcal{AC} := (\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n), \mathcal{AK}, \mathcal{SK})$ is a privacy configuration:
-PC-A: $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \models \mathcal{AK}$.
Lines 2 and 14 of Algorithm 1 reflect the definition of a state cloud as given in Definition 18. Since by this definition $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n) \supseteq \{\Box\psi \mid \psi \in \mathcal{AK}\} = \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(0)$, we have $\mathcal{AK} \subseteq \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$.
-PC-B: $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ are satisfiable as a consequence of credibility.
-PC-C: is obvious, since $\mathcal{SK}$ and $\mathcal{AK}$ are unchanged.

Ad R-B: By effectiveness and Prop 37.
Ad R-A: Let $\mathbf{b} := \mathrm{RTCens}_{(\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n), \mathcal{AK}, \mathcal{SK})}(\mathbf{q})$.
To show: For $1 \leq i \leq n$ it is $a_i = b_i$.
Observe that $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(0) = \mathscr{HC}_{\mathcal{AC},\mathbf{q}}(0) = \bigcup_{\psi \in \mathcal{AK}} \mathrm{Cont}(\psi, t)$. Assume we have checked that $a_k = b_k$ for all $k < i \leq n$. Hence for those k (and especially $k = i-1$)

$$\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(k) = \mathscr{HC}_{\mathcal{AC},\mathbf{q}}(k) \qquad (\star)$$

Case $a_i = t$:
If $\mathscr{HC}_{\mathcal{PC},\mathbf{q}}(i-1) \models q_i$, by $(\star)$ also $\mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \models q_i$. Hence we have $b_i = t$.
Else by $(\star)$: $\mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \cup \mathrm{Cont}(q_i, t)$ and $\mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \cup \mathrm{Cont}(q_i, f)$ do not imply any secret (otherwise already $a_i = r$). Therefore $b_i := \mathsf{eval}(\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n), q_i)$. But $q_i \in \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$, since $\Box q_i \in \mathscr{HC}_{\mathcal{PC},\mathbf{q}}(n)$.
Hence $\mathsf{eval}(\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n), q_i) = t$ and thus $b_i = t$.
Case $a_i = f$: analogous.
Case $a_i = u$: By $(\star)$: $\mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \not\models \Box q_i$ and $\mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \not\models \Box\neg q_i$. Also by $(\star)$

$$\mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \cup \mathrm{Cont}(q_i, t) \not\models \Box\sigma \quad \text{and} \quad \mathscr{HC}_{\mathcal{AC},\mathbf{q}}(i-1) \cup \mathrm{Cont}(q_i, f) \not\models \Box\sigma$$

for any $\sigma \in \mathcal{SK}$. Hence, with $\mathcal{CK} \models \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ (Corollary 38), it follows

$$b_i = \mathsf{eval}(\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n), q_i) = u.$$

Case $a_i = r$: Either $\mathscr{SC}_{\mathcal{PC},\mathbf{q}}(i-1) \cup \{\Box q_i\} \models \sigma$ or $\mathscr{SC}_{\mathcal{PC},\mathbf{q}}(i-1) \cup \{\Box \neg q_i\} \models \sigma$ for a $\sigma \in \mathcal{SK}$. Hence by $(\star)$ either

$$\mathscr{SC}_{\mathcal{AC},\mathbf{q}}(i-1) \cup \{\Box q_i\} \models \sigma \text{ or } \mathscr{SC}_{\mathcal{AC},\mathbf{q}}(i-1) \cup \{\Box \neg q_i\} \models \sigma.$$

Hence $b_i = r$. □

Lemma 47 (Minimally invasive)**.** *The censor* TCens *is minimally invasive.*

Proof. Let $\mathcal{PC}$ be a privacy-configuration, $\mathbf{q}$ a query-sequence and set $\mathbf{a} := \mathrm{TCens}_{\mathcal{PC}}(\mathbf{q})$. Assume there is an index i, s.t. $a_i \neq \mathsf{eval}(\mathcal{KB}, q_i)$. By inspection of the algorithm, this can only be a consequence of line 8 setting $a_i = r$. Hence by line 7 there is a secret such that

$$\mathscr{SC}_{\mathcal{PC},\mathbf{q}}(i-1) \cup \mathrm{Cont}(q_i, \mathsf{eval}(\mathcal{CK}, q_i)) \models \Box\sigma,$$

in violation of effectiveness. □

Lemma 48 (Non-repudiation)**.** *The censor* TCens *is not repudiating.*

Proof. As in the example above, we give a failing setting $\mathcal{PC}$, $\mathbf{q}$.
$\mathcal{CK} := \{\sigma\}$, $\mathcal{AK} := \emptyset$ and $\mathcal{SK} := \{\sigma\}$. $\mathbf{q} := (\sigma, t, \ldots)$. Obviously $a_1 = r$. However in all knowledge-bases $\mathcal{KB}$ s.t. $\mathcal{KB} \not\models \sigma$ the censor would correctly answer either $a_1 = f$ or $a_1 = u$. Hence repudiation fails. □

Corollary 49. *Effectiveness, continuity, credibility and minimal invasion do not imply repudiation.*

The proof of the above lemma can be generalized.

Theorem 50. *A continuous truthful censor satisfies at most two of the properties effectiveness, minimal invasion and repudiation.*

Proof. Assume Cens is continuous, truthful, credible, effective and minimally invasive. We will show that it is not repudiating.
As above, examine the privacy-configuration $\mathcal{PC}$, given by

$$\mathcal{CK} := \{\sigma\}, \mathcal{AK} := \emptyset \quad \text{and} \quad \mathcal{SK} := \{\sigma\},$$

and the query $\mathbf{q} := (\sigma, t, \ldots)$. We set $\mathbf{a} := \mathrm{Cens}(\mathbf{q})$. Obviously $a_1 = r$ must hold, otherwise Cens either lies or reveals a secret. Assume a censored knowledge-base $\mathcal{KB}$ as alternative to $\mathcal{CK}$ and define $\mathbf{a}' := \mathrm{Cens}_{\mathcal{KB},\mathcal{AK},\mathcal{SK}}(\mathbf{q})$. There are three cases:

- $\mathcal{KB} \models \sigma$
- $\mathcal{KB} \models \neg\sigma$
- both $\mathcal{KB} \not\models \sigma$ and $\mathcal{KB} \not\models \neg\sigma$

It suffices to show, that the later two cannot occur.
Assume $\mathcal{KB} \models \neg\sigma$. As consequence of being truthful, the first answer must be either $a'_1 = f$ or $a'_1 = r$. By the fact $\mathscr{SC}_{\mathcal{PC},\mathbf{q}}(0) \cup \{\Box\neg\sigma\} \not\models \Box\sigma$ and minimal invasion (it is the first given answer!) it follows that $a'_1 = f$ and hence we obtain the contradiction to $f = a'_1 \overset{!}{=} a_1 = r$.
Analogous in the third case it follows $a'_1 = u$.
Hence only knowledge-bases that semantically imply σ are possible alternatives to $\mathcal{CK}$, contradicting repudiation. □

3.2 Cooperative Lying Censors

Since the refusing approach did turn out to be unsatisfying, we next want to consider a censor that is capable of lying but not refusing to answer. Formally this means that they are not truthful, but the possible answers are limited to $\mathbb{A} = \{t, f, u\}$. Let us point out that one could adapt all proofs the full answer set (including r) and require that a censor in any situation has an answer different from r. Such a censor is denoted (seemingly) *cooperative*.

In this section we will discuss censors that are minimally invasive, lying and not refusing.

Algorithm 3 Calculate $\mathrm{MILCens}_{\mathcal{PC}}(\mathbf{q})$

Require: $\mathcal{PC} = (\mathcal{CK}, \mathcal{SK}, \mathcal{AK})$ as privacy configuration
Require: $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$
1: $\mathbf{a} = (a_1, a_2, \ldots) \leftarrow (u, u, \ldots)$
2: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(0) \leftarrow \bigcup_{\varphi \in \mathcal{AK}} \mathrm{Cont}(\varphi, t)$
3: **for** $n \leftarrow 1 \ldots \infty$ **do**
4: compliant $\leftarrow$ **true**
5: $p \leftarrow \mathsf{eval}(\mathcal{CK}, q_n)$
6: **for** $\sigma \in \mathcal{SK}$ **do**
7: **if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, p) \models \Box\sigma$ **then**
8: $a_n \leftarrow u$
9: compliant $\leftarrow$ **false**
10: **end if**
11: **end for**
12: **if** compliant **then**
13: $a_n \leftarrow p$
14: **end if**
15: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \leftarrow \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, a_n)$
16: **end for**
17: **return** $\mathbf{a}$

Definition 51 (Minimally invasive lying censor-functions)**.** We denote the censor determined by algorithm 3 as $\mathrm{MILCens}$.

Let us remark that the only difference to TCens is the replacement of the refusal in line 8 with the answer u.

Example 52 (14 cont'd)**.** Calculating the answers of $\mathrm{MILCens}$ in the case where $\mathrm{TheCar} \equiv P$ yields:

$$\mathrm{MILCens}\ldots(\mathbf{P}^1) = (f, f, f, f, u, u, t, t, \ldots)$$
$$\mathrm{MILCens}\ldots(\mathbf{P}^2) = (t, t, t, t, t, u, u, t, t, \ldots)$$

We find that the censor lies to answer on two questions in both sequences. Unsurprisingly the answers refused by TCens are now set to u for the same reasons TCens refused them.

Lemma 53 (Continuity)**.** *The censor* $\mathrm{MILCens}$ *is continuous.*

Proof. Clear by inspection of the algorithm: simply notice that the determination of the answer a_n in lines 8 and 13 only depends on $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1)$ which is determined in the prior loop and the current query q_n. □

Proposition 54. *For all security configurations, query sequences and all $n \in \mathbb{N}_0$:*

$$\textit{if } \Box\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n), \textit{ then } \mathsf{eval}(\mathcal{CK}, \psi) = t.$$

Furthermore $\mathcal{CK} \models \mathcal{A}\mathit{lt}\mathcal{K}_{\mathcal{PC},\mathbf{q}}(n)$.

Proof. By construction in the algorithm, if $\Box\psi \in \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ either

$$\mathsf{eval}(\mathcal{CK}, \psi) = t \;\text{ or }\; \mathsf{eval}(\mathcal{CK}, \neg\psi) = f$$

holds.
In the second case $\neg\psi$ is not satisfiable in $\mathcal{CK}$ and hence $\mathcal{CK} \models \psi$.
Therefore $\mathsf{eval}(\mathcal{CK}, \psi) = t$. □

If $n = 0$ in the above proposition, then $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n)$ encodes exactly the attacker's knowledge and hence the claim trivially holds by the conditions on privacy configurations.

Lemma 55 (Credibility, effectiveness)**.** *The censor* MILCens *is credible and effective.*

Proof. Let $\mathbf{q}, \mathbf{a} := \mathrm{Cens}(\mathbf{q})$ and $n \in \mathbb{N}$. Assume that for all $m < n$ the required properties

(P) $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(m)$ is satisfiable

(E) for all $\sigma \in \mathcal{SK}$ not $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(m) \models \sigma$

hold. We prove that for n these properties hold as well:
In case $\mathcal{SK} = \emptyset$, this is immediate, since the censor will only give true answers and $\mathcal{CK}$ has a model.
Otherwise there are three cases:
First case: Assume $\mathsf{eval}(\mathcal{CK}, q_n) = t$:
There are three sub-cases:
(1): $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \models \Box q_n$: In this case (P) and (E) are immediate.
(2): $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \not\models \Box q_n$ and $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t) \not\models \Box\sigma$ for all $\sigma \in \mathcal{SK}$: Then $a_n = t$ is given by the algorithm and

$$\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) = \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t)$$

is satisfiable, since otherwise in all models (there are none!) all secrets would hold. Hence (P) and (E).
(3): $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \not\models \Box q_n$ and $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t) \models \Box\sigma$ for a $\sigma \in \mathcal{SK}$: Then $a_n = u$ is returned. By Lemma 33 follows (E).
If $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) = \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, u)$ would not be satisfiable, then either $\Box q_n$ or $\Box\neg q_n$ is semantically implied by $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1)$. The first being refused by assumption. If $\Box\neg q_n$ is semantically implied by $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1)$, then by Prop. 37 $\mathcal{A}\mathit{lt}\mathcal{K}_{\mathcal{PC},\mathbf{q}}(n-1) \models \neg q_n$ and hence by Prop. 54 $\mathcal{CK} \models \neg q_n$ contradicting $\mathsf{eval}(\mathcal{CK}, q_n) = t$.

The case $\mathsf{eval}(\mathcal{CK}, q_n) = f$ follows analogous.
The last case $\mathsf{eval}(\mathcal{CK}, q_n) = u$: Obviously $a_n = u$ is returned. Satisfaction of (P) and (E) follows as in **(3)**. □

Lemma 56 (Minimally invasive and lying)**.** *The censor* MILCens *is minimally invasive and lying.*

Proof. Ad "minimally invasive": Assume $\mathbf{a} = \mathrm{MILCens}_{\mathcal{PC}}(\mathbf{q})$ and $a_n \neq \mathsf{eval}(\mathcal{CK}, q_n)$. Then a_n was set in line 8, since by the security check in line 7 effectiveness would have been violated else.
Ad "lying": We give a privacy configuration, a sequence of questions and an index such that the censor will lie:
$\mathcal{CK} := \{\overline{A} \sqcup B \sqsubseteq \overline{C} \sqcup D, A \sqsubseteq B\}$, $\mathcal{SK} := \{C \sqsubseteq D\}$, $\mathcal{AK} := \emptyset$ and $\mathbf{q}$ with
$q_1 := A \sqsubseteq B \to C \sqsubseteq D$, $q_2 := A \sqsubseteq B$ and $q_i := \bot \sqsubseteq \top$ $(i > 2)$.
Will produce $\mathbf{a} := (t, u, t, t, \ldots)$, but $a_2 \notin \{r, t = \mathsf{eval}(\mathcal{CK}, q_2)\}$. □

Lemma 57 (Repudiation)**.** *The censor* MILCens *is repudiating.*

Proof. Let $\mathbf{q}$ be a fixed question series and $\mathbf{a} = \mathrm{MILCens}(\mathbf{q})$.
We show, that $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ is a possible choice of alternate databases:
From Lemma 55 (effectiveness) and since $\{\Box\varphi \mid \varphi \in \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)\} \subseteq \mathscr{K}_{\mathcal{PC},\mathbf{q}}(n)$ it follows that no secret is valid in $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$, hence property R-B).
Ad R-A): observe that each question q_i, where $i \leq n$, exactly one of the following holds:

- $q_i \in \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ iff $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \models q_i$ iff $a_i = t$
- $\neg q_i \in \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ iff $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \models \neg q_i$ iff $a_i = f$
- $q_i, \neg q_i \notin \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ iff $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \not\models q_i$ and $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \not\models \neg q_i$ iff $a_i = u$

This is immediate by credibility and construction.
Hence for all $i \leq n$ $\mathsf{eval}(\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n), q_i) = a_i$.
Furthermore the security condition from line 7 of the algorithm is never satisfied, since otherwise by proposition 37 $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ would violate this condition opposing Lemma 55 (as above).
Therefore all questions q_i, $i \leq n$, are answered by a_i and hence part a).
Ad R-C): Since satisfiability of $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ was shown (PC-B) and neither $\mathcal{AK}$ nor $\mathcal{SK}$ where changed (PC-C), it remains to show $\mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n) \models \mathcal{AK}$ (PC-A). This is immediate, since $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n) \supseteq \{\Box\psi \mid \psi \in \mathcal{AK}\}$ by construction. Therefore $\mathcal{AK} \subseteq \mathcal{AltK}_{\mathcal{PC},\mathbf{q}}(n)$ and hence the proof. □

To finish the chapter, we give a non-effective (and thus also not minimally invasive) but credible censor, that satisfies repudiation. This will prove that repudiation does not imply effectiveness.

Definition 58 (Ineffective repudiating censor)**.** We denote the censor determined by Algorithm 4 as IeRLCens.

Lemma 59. *The censor* IeRLCens *is credible.*

Proof. Observe that only the last else-clause in line 12 in the algorithm can lead to a not satisfiable $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n)$ in line 15: when the algorithm answers within the first two checks the class of models of $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n-1)$ and $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n)$ remains the same. In the next two checks the desired satisfiability of the resulting $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n)$ is an explicit condition.

Concerning the last step, it follows from the first two steps, that both $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \{\Diamond q_n\}$ and $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \{\Diamond \neg q_n\}$ must be satisfiable. Hence by Corollary 33 we conclude that $\mathscr{K}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, u)$ is satisfiable. □

Algorithm 4 Calculate $\mathrm{IeRLCens}_{\mathcal{PC}}(\mathbf{q})$

Require: $\mathcal{PC} = (\mathcal{CK}, \mathcal{SK}, \mathcal{AK})$ as privacy configuration
Require: $\mathbf{q} \in \mathcal{L}^{\mathbb{N}}_{\mathcal{ALC}}$
1: $\mathbf{a} = (a_1, a_2, \ldots) \leftarrow (u, u, \ldots)$
2: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(0) \leftarrow \bigcup_{\varphi \in \mathcal{AK}} \mathrm{Cont}(\varphi, t)$
3: **for** $n \leftarrow 1 \ldots \infty$ **do**
4: **if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \models \mathrm{Cont}(q_n, t)$ **then**
5: $a_n \leftarrow t$
6: **else if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \models \mathrm{Cont}(q_n, f)$ **then**
7: $a_n \leftarrow f$
8: **else if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t)$ is satisfiable **and** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, t) \models \Box\sigma$ for a $\sigma \in \mathcal{SK}$ **then**
9: $a_n \leftarrow t$
10: **else if** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, f)$ is satisfiable **and** $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, f) \models \Box\sigma$ for a $\sigma \in \mathcal{SK}$ **then**
11: $a_n \leftarrow f$
12: **else**
13: $a_n \leftarrow u$
14: **end if**
15: $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(n) \leftarrow \mathscr{H}_{\mathcal{PC},\mathbf{q}}(n-1) \cup \mathrm{Cont}(q_n, a_n)$
16: **end for**
17: **return** $\mathbf{a}$

Lemma 60. *The censor* IeRLCens *is not effective.*

Proof. Consider the privacy configuration $\mathcal{PC}$ given by $\mathcal{KB} := \mathcal{AK} := \emptyset$ and $\mathcal{SK} := \{\sigma\}$ ($\mathcal{KB}, \mathcal{AK}$ can in fact be almost arbitrary).
In this case the query sequence $(\sigma, t, \ldots)$ yields $(t, t, \ldots)$ and hence leads to the privacy violation $\mathscr{H}_{\mathcal{PC},\mathbf{q}}(1) \models \Box\sigma$. □

As a matter of fact, the discussed censor is massively ineffective. It will imply or even confirm a secret whenever it gets a chance without risking its credibility. An option to become "even more" ineffective would be to narrow into a secret, e.g. if a sub-query would be $(\ldots, \psi_1 \wedge \cdots \wedge \psi_n \to \sigma, \psi_1, \ldots \psi_n \ldots)$ the censor should answer t to ψ_1 to ψ_n (if possible), which is not necessarily done by the presented censor. But this would involve a structural analysis of the queried formulae, a feature that we -so far- do not want to equip our censors with. Additionally continuity would have to be dropped.

Lemma 61 (Repudiation)**.** *The censor* IeRLCens *is repudiating.*

Proof. Let $\mathcal{PC} = (\mathcal{CK}, \mathcal{AK}, \mathcal{SK})$ be a privacy configuration and $\mathbf{q}$ be a query-sequence. By construction of the algorithm it is clear that the given answers only depend on $\mathcal{AK}$ and not -by any means- on the actual database.
Hence $\mathcal{KB}(i) := \mathcal{AK}$ is a possible choice as such a sequence.
As remarked, R-A is immediate.
For R-B notice, that by definition of $\mathcal{PC}$ $\mathcal{AK}$ does not validate any secret.
Since $\mathcal{AK} \models \mathcal{AK}$ also follows, that $(\mathcal{AK}, \mathcal{AK}, \mathcal{SK})$ is a privacy-configuration and hence R-C, which completes the proof. □

Let us remark, that the presented censor is only interesting as an example to separate effectiveness and repudiation. A somehow reasonable censor should at least release sometimes "new" information (i.e. not known by the attacker yet) from the protected knowledge-base. In the above setting, the attacker only can learn the potential secrets in case it did not know them already. The censor is also extremely far from being minimally invasive.

3.3 Combined Censors

A major difference between censors for incomplete propositional databases and our censors for $\mathcal{ALC}$ knowledge-bases is that in our case answering u cannot reveal any harmful information to a querying attacker. We made use of this to present a cooperative lying censor that has all desired properties like minimal invasiveness. In the propositional case, the disjunction of all secrets has to be added as an extra secret when lying only censors are considered. This leads, of course, to the problem that no censor can protect a formula and its negation at the same time. Even more problematic is that this disjunction must not be inferable from the attacker's pre-knowledge.

Thus in the propositional case censors that are both lying and refusing are often needed. In our case this is not necessary since there is a lying only censor that has all the nice properties.

4 Conclusion and Future Work

The presented work formalizes censors that can hide actual knowledge within incomplete Boolean systems constructed on top of $\mathcal{ALC}$.

However, our approach covers only a subset of the expressive power of $\mathcal{ALC}$, namely conceptual subsumption. In a censored inclomplete $\mathcal{ALC}$ system, we should also be able to deal with retrieval queries, i.e. queries that ask for all individuals that belong to a given concept. In this situation a censor should not only protect subsumptional and coherence information, but also the actual domains of the concepts as well as specific individuals belonging to a concept. So far it is not clear how in such a setting secrets can be specified and how protection mechanisms can be formalized and dealt with.

Another question is how to adapt censoring systems when censors should be able to hide ignorance, i.e. they should hide that a given fact is not inferable nor refutable from a given knowledge-base.

We believe that the presented formal definitions and methods can be extended to deal with retrieval queries as well as to systems with new types of secrets such as hiding unknowlege.

Acknowledgements

We would like to thank the anonymous referees whose comments helped to substantially improve the paper.

References

[1] Carlos Areces, Patrick Blackburn, Bernadette Martinez Hernandez, and Maarten Marx. Handling Boolean ABoxes. In *Proc. of the 2003 Description Logic Workshop*. CEUR (http://ceur-ws.org), 2003.

[2] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, New York, NY, USA, 2003.

[3] Franz Baader, Silvio Ghilardi, and Carsten Lutz. LTL over description logic axioms. *ACM Transactions Computational Logic*, 13(3):21:1–21:32, August 2012.

[4] Franz Baader and Werner Nutt. Basic description logics. In Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors, *The Description Logic Handbook*, pages 43–95. Cambridge University Press, 2003.

[5] Jie Bao, Giora Slutzki, and Vasant Honavar. Privacy-preserving reasoning on the semantic web. In *Web Intelligence 2007*, pages 791–797, 2007.

[6] Joachim Biskup. For unknown secrecies refusal is better than lying. *Data & Knowledge Engineering*, 33(1):1–23, 2000.

[7] Joachim Biskup and Piero A. Bonatti. Controlled query evaluation for enforcing confidentiality in complete information systems. *International Journal of Information Security*, 3(1):14–27, 2004.

[8] Joachim Biskup and Piero A. Bonatti. Controlled query evaluation for known policies by combining lying and refusal. *Annals of Mathematics and Artificial Intelligence*, 40(1-2):37–62, 2004.

[9] Joachim Biskup and Piero A. Bonatti. Controlled query evaluation with open queries for a decidable relational submodel. *Annals of Mathematics and Artificial Intelligence*, 50(1-2):39–77, 2007.

[10] Joachim Biskup and Torben Weibert. Keeping secrets in incomplete databases. *International Journal of Information Security*, 7(3):199–217, May 2008.

[11] Piero A. Bonatti, Sarit Kraus, and V. s. Subrahmanian. Foundations of secure deductive databases. *Transactions on Knowledge and Data Engineering*, 7(3):406–422, 1995.

[12] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. View-based query answering over description logic ontologies. In *Principles of Knowledge Representation and Reasoning*, pages 242–251, 2008.

[13] Bernardo Cuenca Grau and Ian Horrocks. Privacy-preserving query answering in logic-based information systems. In *18th European Conference on Artificial Intelligence (ECAI-2008)*. IOS Press, 2008.

[14] Bernardo Cuenca Grau, Evgeny Kharlamov, Egor Kostylev, and Dmitriy Zheleznyakov. Controlled query evaluation over lightweight ontologies. In *Proceedings of the 27th International Workshop on Description Logics (DL)*, pages 141–152, 2014.

[15] Hongkai Liu, Carsten Lutz, Maja Milii, and Frank Wolter. Updating description logic ABoxes. In *International Conference of Principles of Knowledge Representation and Reasoning*, pages 46–56. American Association for Artificial Intelligence (http://www.aaai.org), 2006.

[16] Carsten Lutz, Holger Sturm, Frank Wolter, and Michael Zakharyaschev. A tableau decision algorithm for modalized ALC with constant domains. *Studia Logica*, 72(2):199–232, 2002.

[17] Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. Temporal description logics: a survey. In *Proceedings of the 15th International Symposium on Temporal Representation and Reasoning TIME 08*, pages 3–14. IEEE Computer Society, 2008.

[18] Gerome Miklau and Dan Suciu. A formal analysis of information disclosure in data exchange. *Journal of Computer and System Sciences*, 73(3):507–534, 2007.

[19] George L. Sicherman, Wiebren De Jonge, and Reind P. Van de Riet. Answering queries without revealing secrets. *ACM Transactions on Database Systems*, 8(1):41–59, 1983.

[20] Kilian Stoffel and Thomas Studer. Provable data privacy. In K. Viborg, J. Debenham, and R. Wagner, editors, *Database and Expert Systems Applications (DEXA 2005)*, volume 3588 of *LNCS*, pages 324–332. Springer, 2005.

[21] Phiniki Stouppa and Thomas Studer. Data privacy for $\mathcal{ALC}$ knowledge bases. In S. Artemov

and A. Nerode, editors, *Logical Foundations of Computer Science (LFCS 2009)*, volume 5407 of *LNCS*, pages 409–421. Springer, 2009.

[22] Thomas Studer. Privacy preserving modules for ontologies. In A. Pnueli, I. Virbitskaite, and A. Voronkov, editors, *Perspectives of System Informatics (PSI'09)*, volume 5947 of *LNCS*, pages 380–387. Springer, 2010.

[23] Thomas Studer. Justified terminological reasoning. In E. Clarke, I. Virbitskaite, and A. Voronkov, editors, *Perspectives of System Informatics (PSI'11)*, volume 7162 of *LNCS*, pages 349–361. Springer, 2012.