Database Fragmentation with Encryption: Under Which Semantic Constraints and A Priori Knowledge Can Two Keep a Secret?

Joachim Biskup, Marcel Preuß

Database Fragmentation with Encryption: Under Which Semantic Constraints and A Priori Knowledge Can Two Keep a Secret?*

Joachim Biskup and Marcel Preuß

Technische Universität Dortmund, Dortmund, Germany
{biskup,preuss}@ls6.cs.tu-dortmund.de

Abstract. Database outsourcing to semi-honest servers raises concerns against the confidentiality of sensitive information. To hide such information, an existing approach splits data among two supposedly mutually isolated servers by means of fragmentation and encryption. This approach is modelled logic-orientedly and then proved to be confidentiality preserving, even if an attacker employs some restricted but nevertheless versatile class of a priori knowledge to draw inferences. Finally, a method to compute a secure fragmentation schema is developed.

Keywords: A Priori Knowledge, Confidentiality Constraint, Fragmentation, Inference-Proofness, Logic, Outsourcing, Semi-Honest Server.

1 Introduction

Database outsourcing faces two directly conflicting goals: it should both reduce storage and processing costs by storing data on external servers as well as provably comply with confidentiality requirements – in particular with privacy concerns – in spite of storing data externally [10]. A basic solution presented in [2,9] aims at resolving this conflict by means of the combined usage of fragmentation and encryption: a client's database relation is losslessly decomposed into (at least) two vertical fragments each of which is maintained by a different semi-honest server; sensitive data is split into harmless parts, either by breaking an association or by separating an encrypted piece of data from the cryptographic key employed; moreover, the servers are (postulated to be) mutually isolated and each attacker is assumed to have access to at most one server.

Consequently, due to splitting, each attacker (identified with a server) only has accesses to non-sensitive data and, due to losslessness, an authorized user (identified with the client) can still reconstruct the original data while, due to isolation, only authorized users can do so.

Example 1. We consider the relational instance about medical data shown in the upper half of Fig. 1. Suppose that social security numbers (`SSN`) should be hidden, as well as associations between a patient identified by his name (`Name`)

* This work has been supported by the DFG under grant SFB 876/A5.

R	SSN	Name	Illness	HurtBy	Doctor
	1234	Hellmann	Borderline	Hellmann	White
	2345	Dooley	Laceration	McKinley	Warren
	3456	McKinley	Laceration	Dooley	Warren
	3456	McKinley	Concussion	Dooley	Warren

F_1	tid	SSN	Name	HurtBy	Doctor
	1	e_S^1	Hellmann	e_H^1	White
	2	e_S^2	Dooley	e_H^2	Warren
	3	e_S^3	McKinley	e_H^3	Warren
	4	e_S^4	McKinley	e_H^4	Warren

F_2	tid	SSN	HurtBy	Illness
	1	κ_S^1	κ_H^1	Borderline
	2	κ_S^2	κ_H^2	Laceration
	3	κ_S^3	κ_H^3	Laceration
	4	κ_S^4	κ_H^4	Concussion

Fig. 1. A relational instance containing sensitive data items and associations together with a possible fragmentation with encryption

and an illness treated (`Illness`), between a patient (`Name`) and a person who caused an illness (`HurtBy`), and between an illness (`Illness`) and a person having caused that illness (`HurtBy`), respectively. The lower half of Fig. 1 exhibits a possible fragmentation with encryption: The sensitive association between `Name` and `Illness` is "broken" by separating the attribute `Name` in the fragment F_1 from the attribute `Illness` in the fragment F_2. The sensitive associations between `Name` and `HurtBy` and between `Illness` and `HurtBy` are made "invisible" by using encryption for the attribute `HurtBy` such that ciphertexts are stored in fragment F_1 and corresponding keys in fragment F_2. The sensitive attribute `SSN` is similarly treated by encryption. The newly introduced tuple identifiers (`tid`) ensure the losslessness of the vertical decomposition (see, e.g., [1]).

At first glance two semi-honest servers seem to "keep the secrets" declared in a confidentiality policy. However, a second thought raises some doubts on the actual achievements: though each server only stores data that is non-sensitive per se, an attacker might still be able to infer sensitive information by exploiting his a priori knowledge obtained from further sources. In particular, this a priori knowledge might comprise semantic constraints to be satisfied by the relation being decomposed and individual fact data stemming from the "outside world".

Example 2. Suppose an attacker has access to the fragment F_1 and knows a priori that Doctor `White` is a psychiatrist only treating patients suffering from the `Borderline`-syndrome. The attacker can then conclude that patient `Hellmann` suffers from the illness `Borderline`-syndrome, thereby violating the requirement that associations between a patient and an illness treated should be hidden. Moreover, if this attacker additionally knows that all patients suffering from the `Borderline`-syndrome have hurt themselves, the attacker can conclude that patient `Hellmann` has been hurt by `Hellmann`, thereby revealing an association between a patient and a person who caused an illness.

The first violation is enabled by a priori knowledge connecting a fact shown in the visible fragment with a fact in the hidden fragment, namely by means of the *constant symbols* `White` and `Borderline`. Similarly, the second violation is caused by a priori knowledge that connects two concepts across the decomposition, namely the concept of a patient and the concept of a hurt creator, where

a concept will be formally represented by a *variable* ranging over the domain of an attribute. Such connections might "transfer information" between the visible fragment and the hidden fragment. In other words, an attacker a priori knowing such connections might infer hidden information from visible information. Next, we introduce a more abstract example in a more formal way.

Example 3. The client maintains a relational schema with relational symbol R, attribute set $A_R = \{a_1, a_2, a_3, a_4\}$ and the functional dependency $a_2 \rightarrow a_3$ as a semantic constraint. Confidentiality interests are expressed by a set $\mathcal{C} = \{\{a_1, a_3\}, \{a_4\}\}$ of two confidentiality constraints: $\{a_1, a_3\}$ is intended to require to hide the associations between values of the attributes a_1 and a_3, and $\{a_4\}$ requires to hide single values of attribute a_4. The a priori knowledge comprises the functional dependency and a sentence expressing the following: "for some specific values b and c for the attributes a_2 and a_3, resp., there exist a value X_1 for attribute a_1 and a value X_4 for attribute a_4 such that the tuple $(a_1 : X_1\,,\ a_2 : b\,,\ a_3 : c\,,\ a_4 : X_4)$ is an element of the relational instance r". Furthermore, fragment F_1 has attribute set $A_{F_1} = \{tid, a_1, a_2, a_4\}$ and fragment F_2 attribute set $A_{F_2} = \{tid, a_3, a_4\}$ such that the common attribute a_4 is encrypted.

Let fragment F_1 exhibit a tuple $(tid : no\,,\ a_1 : a\,,\ a_2 : b\,,\ a_4 : ran)$, where *no* is a tuple identifier and *ran* results from encryption. Combining the a priori knowledge with the tuple exhibited, an attacker might infer that the value a for attribute a_1 is associated with the value c for attribute a_3, thereby violating the confidentiality constraint $\{a_1, a_3\}$. Thus fragment F_1 is not inference-proof under the given assumptions. In contrast, fragment F_2 is harmless.

The a priori knowledge relates the fragments F_1 and F_2 by means of both the functional dependency using variables and the association fact about a and b dealing with constant symbols. Though taken alone, each of these items might be harmless, their combination turns out to be potentially harmful. The next example indicates that for the same underlying situation one fragmentation satisfying required confidentiality constraints might be better than another one.

Example 4. Modifying Example 3 such that $A_{F_1} = \{tid, a_1, a_4\}$ and $A_{F_2} = \{tid, a_2, a_3, a_4\}$ would block the harmful inference. For, intuitively, the crucial fact about the association of a with b does not span across the decomposition.

More generally, we will investigate the following problems in this article:

- Given a fragmentation, identify conditions on the a priori knowledge to provably disable an attacker to infer sensitive information.
- Given some a priori knowledge, determine a fragmentation such that an attacker cannot infer sensitive information.

Our solutions will be based on a logic-oriented modelling of the fragmentation approach presented in [2,9] within the more general framework of Controlled Interaction Execution, CIE, as surveyed in [3]. This framework assists a database owner in ensuring that each of his interaction partners can only obtain a dedicated inference-proof view on the owner's data: each of these views does not

contain information to be kept confidential from the respective partner, even if this partner tries to employ inferences by using his a priori knowledge and his general awareness of the protection mechanism. Our main achievements can be summarized as follows and will be elaborated in the remainder as indicated:

- We formalize the fragmentation approach of [2,9] (Sect. 2).
- We provide a logic-oriented modelling of that approach (Sect. 3).
- We exhibit sufficient conditions to achieve confidentiality (Sect. 4).
- We propose a method to compute a suitable fragmentation (Sect. 5).

These results extend the previous work [5] in which a more simple approach to fragmentation proposed in [7] – splitting a relational instance into one externally stored part and one locally-held part without resorting to encryption – is formally analyzed to be inference-proof. In particular, the previous work is extended by a more detailed formal modelling of fragmentation including encryption of values, a more expressive class of sentences representing an attacker's a priori knowledge and a method to compute an inference-proof fragmentation.

2 Confidentiality by Fragmentation

In this section, we briefly formalize and extend the approach to fragmentation proposed in [2,9]. All data is represented within a single relational instance r over a relational schema $\langle R|A_R|SC_R\rangle$ with relational symbol R and the set $A_R = \{a_1, \ldots, a_n\}$ of attributes, for simplicity assumed to have the same type given by the infinite set $\mathcal{U}$ of values. Moreover, the set SC_R contains some semantic (database) constraints, which must be satisfied by the relational instance r.

The idea for achieving confidentiality basically lies in splitting the original instance r vertically (i.e., by projections on subsets of A_R) into two fragment instances f_1 and f_2 each of which is stored on *exactly one* of the two external servers instead of r. Those confidentiality requirements which cannot be satisfied by just splitting instance r are satisfied by encrypting the values of some attributes. Each "encrypted attribute" is contained in f_1 – storing ciphertexts – as well as in f_2 – storing globally unique cryptographic keys.

We assume an encryption function $Enc : \mathcal{U} \times \mathcal{U} \to \mathcal{U}$ satisfying the group properties to achieve perfect (information-theoretic) security. A value of $\mathcal{U}$ might be used not only as a plaintext but also as a cryptographic key and a ciphertext. The decryption function is defined by $Dec(e, \kappa) = v$ iff $Enc(v, \kappa) = e$.

Definition 1 (Fragmentation). *Given a relational schema $\langle R|A_R|SC_R\rangle$, a* vertical fragmentation *$(\mathcal{F}, \mathcal{E})$ of $\langle R|A_R|SC_R\rangle$ contains a set $\mathcal{E} \subseteq A_R$ of so-called "encrypted attributes" and a set $\mathcal{F} = \{\langle F_1|A_{F_1}|SC_{F_1}\rangle, \langle F_2|A_{F_2}|SC_{F_2}\rangle\}$ in which $\langle F_1|A_{F_1}|SC_{F_1}\rangle$ and $\langle F_2|A_{F_2}|SC_{F_2}\rangle$ are relational schemas called* fragments *of $(\mathcal{F}, \mathcal{E})$ both containing the distinguished attribute $a_{tid} \notin A_R$ for tuple identifiers. Moreover, for $i \in \{1, 2\}$, it holds that*

(i) $A_{F_i} := \{a_{tid}\} \cup \bar{A}_{F_i}$ *with* $\bar{A}_{F_i} \subseteq A_R$,

	$A_{F_i} \setminus A_R$	$(A_{F_1} \setminus \mathcal{E}) \cap A_R$	$\mathcal{E} \cap A_{F_i} \cap A_R$	$(A_{F_2} \setminus \mathcal{E}) \cap A_R$
A_R		$a_1, \ldots, a_h$	$a_{h+1}, \ldots, a_k$	$a_{k+1}, \ldots, a_n$
A_{F_1}	$a_{\mathtt{tid}}$	$a_1, \ldots, a_h$	$a_{h+1}, \ldots, a_k$	
A_{F_2}	$a_{\mathtt{tid}}$		$a_{h+1}, \ldots, a_k$	$a_{k+1}, \ldots, a_n$

Fig. 2. Rearrangement of columns of r, f_1 and f_2

(ii) $SC_{F_i} := \{a_{tid} \rightarrow \bar{A}_{F_i}\}$ *with* $a_{tid} \rightarrow \bar{A}_{F_i}$ *being a functional dependency declaring* a_{tid} *as a primary key,*
(iii) $\bar{A}_{F_1} \cup \bar{A}_{F_2} = A_R$ *and* $\bar{A}_{F_1} \cap \bar{A}_{F_2} = \mathcal{E}$.

Given a relational instance r *over* $\langle R|A_R|SC_R\rangle$*, the* fragment instances f_1 *and* f_2 *over* $\langle F_1|A_{F_1}|SC_{F_1}\rangle$ *and* $\langle F_2|A_{F_2}|SC_{F_2}\rangle$ *are created by inserting exactly both the tuples* ν_1 *into* f_1 *and* ν_2 *into* f_2 *for each tuple* $\mu \in r$*. Thereby,*

(a) $\nu_1[a_{tid}] = \nu_2[a_{tid}] = v_\mu$ *s.t.* v_μ *is a globally unique tuple identifier,*
(b) $\nu_i[a] = \mu[a]$ *for* $i \in \{1, 2\}$ *and for each attribute* $a \in (\bar{A}_{F_i} \setminus \mathcal{E})$*,*
(c) $\nu_1[a] := Enc(\mu[a], \kappa)$ *and* $\nu_2[a] := \kappa$ *for each* $a \in \mathcal{E}$ *s.t.* κ *is a cryptographic key being random but globally unique for each value of each tuple.*

W.l.o.g. we suppose that $A_R := \{a_1, \ldots, a_h, a_{h+1}, \ldots, a_k, a_{k+1}, \ldots, a_n\}$ is the set of attributes of $\langle R|A_R|SC_R\rangle$ and that the columns of the instances r, f_1 and f_2 are rearranged as visualized in Fig. 2. The columns $h+1, \ldots, k$ differ in the interpretation of the values stored in the instances r, f_1 and f_2: although each of the tuples $\mu \in r$, $\nu_1 \in f_1$ and $\nu_2 \in f_2$ assign values to the attributes $a_{h+1}, \ldots, a_k$, $\mu[a_j]$ is a plaintext value, $\nu_1[a_j]$ is a ciphertext value and $\nu_2[a_j]$ is a cryptographic key. In contrast, for $a_1, \ldots, a_h$ ($a_{k+1}, \ldots, a_n$, respectively) corresponding tuples of r and f_1 (r and f_2) share the same combination of values.

To enable an authorized user having access to both fragment-instances f_1 and f_2 to query all information contained in the original instance r, fragmentation ensures that in f_1 and f_2 exactly those two tuples $\nu_1 \in f_1$ and $\nu_2 \in f_2$ corresponding to a tuple of r share the same unique tuple ID (item (a) of Def. 1). Thus, if $\nu_1[a_{\mathtt{tid}}] = \nu_2[a_{\mathtt{tid}}]$, two tuples $\nu_1 \in f_1$ and $\nu_2 \in f_2$ can be recomposed to a tuple of r with the help of a binary operation denoted by $\diamond$.

As the goal is to achieve confidentiality by fragmentation, a formal declaration of confidentiality requirements is indispensable. In [2,9] this is obtained by defining a set of so-called confidentiality constraints on schema level.

Definition 2 (Confidentiality Constraint). *A* confidentiality constraint c *over a relational schema* $\langle R|A_R|SC_R\rangle$ *is a non-empty subset* $c \subseteq A_R$.

Semantically, a confidentiality constraint c claims that each combination of values allocated to the set $c \subseteq A_R$ of attributes in the original instance r over schema $\langle R|A_R|SC_R\rangle$ should neither be contained completely in the unencrypted part of f_1 nor be contained completely in the unencrypted part of f_2.

Definition 3 (Confidentiality of Fragmentation). *Let* $\langle R|A_R|SC_R\rangle$ *be a relational schema,* $(\mathcal{F}, \mathcal{E})$ *a fragmentation of* $\langle R|A_R|SC_R\rangle$ *according to Def. 1 and* $\mathcal{C}$ *a set of confidentiality constraints over* $\langle R|A_R|SC_R\rangle$ *according to Def. 2.* $(\mathcal{F}, \mathcal{E})$ *is* confidential *w.r.t.* $\mathcal{C}$ *iff* $c \not\subseteq (A_{F_1} \setminus \mathcal{E})$ *and* $c \not\subseteq (A_{F_2} \setminus \mathcal{E})$ *for each* $c \in \mathcal{C}$.

Example 5. The fragmentation depicted in Fig. 1 is confidential w.r.t. the set $\mathcal{C} = \{c_1, c_2, c_3, c_4\}$ of confidentiality constraints such that $c_1 = \{\texttt{SSN}\}$, $c_2 = \{\texttt{Name}, \texttt{Illness}\}$, $c_3 = \{\texttt{Name}, \texttt{HurtBy}\}$, and $c_4 = \{\texttt{Illness}, \texttt{HurtBy}\}$.

3 A Logic-Oriented View on Fragmentation

In this section we will present a logic-oriented modelling of fragmentation, for conciseness mostly focussing on the attacker's point of view resulting from his knowledge of the fragment instance f_1, which is supposed to be known to him.

To set up the universe of discourse, we start by defining the set $\mathcal{P}$ of predicate symbols of a language $\mathscr{L}$ of first-order logic with equality. First, to model the attacker's knowledge about the fragment instance f_1, we need the predicate symbol $F_1 \in \mathcal{P}$ with arity $k + 1 = |A_{F_1}|$ (including the additional tuple ID attribute plus k original attributes (cf. Fig. 2)). Second, to capture the attacker's awareness of the fragmentation, in particular his partial knowledge about the hidden original instance r and the separated second fragmentation instance f_2, we additionally use the predicate symbols R with arity $n = |A_R|$ and F_2 with arity $n - h + 1 = |A_{F_2}|$. Additionally, the distinguished predicate symbol $\equiv \notin \mathcal{P}$ is available in $\mathscr{L}$ for expressing equality.

We employ the binary function symbols E and D for modelling the attacker's knowledge about the encryption function *Enc* and the inverse decryption function *Dec*. Finally, we denote tuple values by elements of the set *Dom* of constant symbols, which will be employed as the universe of (Herbrand) interpretations for $\mathscr{L}$ as well. In compliance with CIE (e.g., [4,6]) this set is assumed to be fixed and infinite. Further, we have an infinite set *Var* of variables.

As usual, the formulas contained in $\mathscr{L}$ are constructed inductively using the quantifiers $\forall$ and $\exists$ and the connectives $\neg$, $\wedge$, $\vee$ and $\Rightarrow$. Closed formulas, i.e., formulas without free occurrences of variables, are called sentences. This syntactic specification is complemented with a semantics which reflects the characteristics of databases by means of so-called DB-Interpretations according to [4,6]:

Definition 4 (DB-Interpretation). *Given the language $\mathscr{L}$ described above, an interpretation $\mathcal{I}$ over a universe $\mathcal{U}$ is a* DB-Interpretation *for $\mathscr{L}$ iff*

(i) $\mathcal{U} := \mathcal{I}(Dom) = Dom$,
(ii) $\mathcal{I}(v) = v \in \mathcal{U}$ *for every constant symbol* $v \in Dom$,
(iii) $\mathcal{I}(E)(v, \kappa) = e$ *iff* $Enc(v, \kappa) = e$, *for all* $v, \kappa, e \in \mathcal{U}$,
(iv) $\mathcal{I}(D)(e, \kappa) = v$ *iff* $Dec(e, \kappa) = v$, *for all* $v, \kappa, e \in \mathcal{U}$,
(v) every $P \in \mathcal{P}$ *with arity* m *is interpreted by a finite relation* $\mathcal{I}(P) \subset \mathcal{U}^m$,
(vi) the predicate symbol $\equiv \notin \mathcal{P}$ *is interpreted by* $\mathcal{I}(\equiv) = \{(v, v) \mid v \in \mathcal{U}\}$.

If item (v) is instantiated by taking the instances r, f_1 and f_2 as interpretations of P=R, F_1, and F_2, respectively, the resulting DB-Interpretation $\mathcal{I}_{r,f_1,f_2}$ – or just $\mathcal{I}_r$ for short if f_1 and f_2 are derived from r according to Def. 1 – is called induced *by r (and f_1 and f_2).*

The notion of *satisfaction/validity* of formulas in $\mathscr{L}$ by a DB-Interpretation is the same as in usual first-order logic. A set $\mathcal{S} \subset \mathscr{L}$ of sentences *implies/entails* a sentence $\Phi \in \mathscr{L}$ (written as $\mathcal{S} \models_{DB} \Phi$) iff each DB-Interpretation $\mathcal{I}$ satisfying $\mathcal{S}$ (written as $\mathcal{I} \models_M \mathcal{S}$) also satisfies Φ (written as $\mathcal{I} \models_M \Phi$).

Considering an attacker knowing the fragment instance f_1, the attacker's *positive* knowledge about the tuples explicitly recorded in f_1 can be simply modelled logic-orientedly by adding an atomic sentence $F_1(\nu[a_{\mathtt{tid}}], \nu[a_1], \ldots, \nu[a_k])$ for each tuple $\nu \in f_1$. As the original instance r – and so its fragment instance f_1 – is assumed to be *complete*[1], each piece of information expressible in $\mathscr{L}$ which is *not* contained in r (f_1, resp.) is considered to be *not* valid by Closed World Assumption (CWA). The concept of DB-Interpretations fully complies with the semantics of *complete* relational instances. Accordingly, an attacker knows that each of the infinite combinations of values $(v_{\mathtt{tid}}, v_1, \ldots, v_k) \in Dom^{k+1}$ not contained in any tuple of f_1 leads to a valid sentence $\neg F_1(v_{\mathtt{tid}}, v_1, \ldots, v_k)$.

As this *negative* knowledge is not explicitly enumerable, it is expressed implicitly by a so-called completeness sentence (cf. [4]) having a universally quantified variable X_j for each attribute $a_j \in A_{F_1}$ (sentence (2) of Def. 5 below). This completeness sentence expresses that every constant combination $(v_{\mathtt{tid}}, v_1, \ldots, v_k) \in Dom^{k+1}$ (substituting the universally quantified variables $X_{\mathtt{tid}}, X_1, \ldots X_k$) either appears in f_1 or satisfies the sentence $\neg F_1(v_{\mathtt{tid}}, v_1, \ldots, v_k)$. By construction, this completeness sentence is satisfied by any DB-Interpretation induced by f_1.

Example 6. For the medical example, the knowledge implicitly taken to be *not* valid by CWA can be expressed as the following completeness sentence:

$$
\begin{array}{l}
(\forall X_t)(\forall X_S)(\forall X_N)(\forall X_H)(\forall X_D)\ [\\
(X_t \equiv 1 \ \wedge\ X_S \equiv e_S^1 \ \wedge\ X_N \equiv \text{Hellmann} \ \wedge\ X_H \equiv e_H^1 \ \wedge\ X_D \equiv \text{White}) \ \vee \\
(X_t \equiv 2 \ \wedge\ X_S \equiv e_S^2 \ \wedge\ X_N \equiv \text{Dooley} \ \wedge\ X_H \equiv e_H^2 \ \wedge\ X_D \equiv \text{Warren}) \ \vee \\
(X_t \equiv 3 \ \wedge\ X_S \equiv e_S^3 \ \wedge\ X_N \equiv \text{McKinley} \ \wedge\ X_H \equiv e_H^3 \ \wedge\ X_D \equiv \text{Warren}) \ \vee \\
(X_t \equiv 4 \ \wedge\ X_S \equiv e_S^4 \ \wedge\ X_N \equiv \text{McKinley} \ \wedge\ X_H \equiv e_H^4 \ \wedge\ X_D \equiv \text{Warren}) \ \vee \\
\neg F_1(X_t, X_S, X_N, X_H, X_D) \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad]
\end{array}
$$

Based on the explanations given so far, an attacker's knowledge about the fragment instance f_1 can be formalized logic-orientedly as follows:

Definition 5 (Logic-Oriented View on f_1). *Given a fragment instance f_1 over $\langle F_1 | A_{F_1} | SC_{F_1} \rangle$ according to Def. 1 with $A_{F_1} = \{a_{\mathtt{tid}}, a_1, \ldots, a_k\}$, the positive knowledge contained in f_1 is modelled in $\mathscr{L}$ by the set of sentences*

$$db_{f_1}^{+} := \{F_1(\nu[a_{\mathtt{tid}}], \nu[a_1], \ldots, \nu[a_k]) \mid \nu \in f_1\} \ . \tag{1}$$

The implicit negative knowledge contained in f_1 is modelled in $\mathscr{L}$ by the singleton set $db_{f_1}^{-}$ containing the completeness sentence

$$(\forall X_{\mathtt{tid}}) \ldots (\forall X_k) \left[\bigvee_{\nu \in f_1} \left(\bigwedge_{a_j \in A_{F_1}} (X_j \equiv \nu[a_j]) \right) \vee \neg F_1(X_{\mathtt{tid}}, X_1, \ldots, X_k) \right] . \tag{2}$$

[1] Though not explicitly stated in [2,9], in this article we follow the usual intuitive semantics of complete instances.

Moreover the functional dependency $a_{tid} \rightarrow \{a_1, \ldots, a_k\} \in SC_{F_1}$ *is modelled in* $\mathscr{L}$ *by the singleton set* fd_{F_1} *containing the sentence*

$$\begin{aligned}(\forall X_{tid})\,(\forall X_1)\ldots(\forall X_k)\,(\forall X'_1)\ldots(\forall X'_k)\,[\,F_1(X_{tid}, X_1, \ldots, X_k) \wedge \\ F_1(X_{tid}, X'_1, \ldots, X'_k) \Rightarrow (X_1 \equiv X'_1) \wedge \ldots \wedge (X_k \equiv X'_k)\,]\ .\end{aligned} \tag{3}$$

Overall the logic-oriented view on f_1 *in* $\mathscr{L}$ *is* $db_{f_1} := db^+_{f_1} \cup db^-_{f_1} \cup fd_{F_1}$.

Proposition 1. *Under the assumptions of Def. 5, the sentences (1), (2) and (3) of* db_{f_1} *are satisfied by the DB-interpretation* $\mathcal{I}_r$*, i.e.,* $\mathcal{I}_r \models_M db_{f_1}$.

Proof. Direct consequence of the definitions. □

An attacker is assumed to know the process of fragmentation as well as the schemas $\langle R|A_R|SC_R\rangle$ and $\langle F_2|A_{F_2}|SC_{F_2}\rangle$ of the instances kept hidden from him. Thus he can infer that for each tuple $\nu_1 \in f_1$ there are tuples $\nu_2 \in f_2$ and $\mu \in r$ satisfying the equation $\nu_1 \diamond \nu_2 = \mu$. So, an attacker knows all values assigned to the set $(A_{F_1} \setminus \mathcal{E}) \cap A_R$ of unencrypted attributes in μ from his knowledge of ν_1, whereas in general he only knows the existence of values for the remaining attributes (sentence (4) of Def. 6 below). Similarly, the attacker is not able to infer the cleartext values assigned to the attributes of $\mathcal{E}$ in μ: by the group properties of the encryption function, each ciphertext considered might be mapped to each possible cleartext without knowing the specific key hidden in fragment f_2.

Next, an attacker knows that a tuple $\nu_2 \in f_2$ can only exist if also corresponding tuples $\nu_1 \in f_1$ and $\mu \in r$ satisfying the equation $\nu_1 \diamond \nu_2 = \mu$ exist (sentence (5) of Def. 6 below). According to the if-part of sentence (6) this requirement analogously holds for the existence of each tuple of r. The only-if-part of sentence (6) describes the fact that the (hypothetical) knowledge of both tuples $\nu_1 \in f_1$ and $\nu_2 \in f_2$ with $\nu_1[a_{\mathtt{tid}}] = \nu_2[a_{\mathtt{tid}}]$ would enable the attacker to reconstruct the tuple $\mu \in r$ satisfying $\mu = \nu_1 \diamond \nu_2$ completely.

Based on the one-to-one correspondence between each tuple $\mu \in r$ and a tuple $\nu_1 \in f_1$ ($\nu_2 \in f_2$, resp.), observing that two different tuples $\nu_1, \nu'_1 \in f_1$ are equal w.r.t. the values allocated to the unencrypted attributes of $(A_{F_1} \setminus \mathcal{E}) \cap A_R$, an attacker can reason that there are also two tuples $\mu, \mu' \in r$ which are equal w.r.t. the values allocated to these attributes, but differ in at least one of the values allocated to $A_R \setminus (A_{F_1} \setminus \mathcal{E})$ (sentence (7) (sentence (8) in case of f_2) of Def. 6 below). Otherwise, the instance r would have duplicates.

Summarizing, and for now neglecting semantic constraints, an attacker's logic-oriented view on the (hidden) instances r and f_2 can be modelled as follows:

Definition 6 (Fragmentation Logic-Oriented). *Let* $(\mathcal{F}, \mathcal{E})$ *be a fragmentation of a relational schema* $\langle R|A_R|SC_R\rangle$ *with instance* r *and let* f_1 *and* f_2 *be the corresponding fragment instances over the fragments* $\langle F_1|A_{F_1}|SC_{F_1}\rangle \in \mathcal{F}$ *and* $\langle F_2|A_{F_2}|SC_{F_2}\rangle \in \mathcal{F}$ *according to Def. 1.*

The knowledge about r and f_2 deduced from the knowledge of f_1 is expressed by

$$\begin{array}{l}(\forall X_{tid})\,(\forall X_1)\ldots(\forall X_h)\,(\forall X_{h+1})\ldots(\forall X_k)\,\big[\\ \quad F_1\,(X_{tid},X_1,\ldots,X_h,X_{h+1},\ldots,X_k)\\ \quad\quad\Rightarrow\\ \quad(\exists Y_{h+1})\ldots(\exists Y_k)\,(\exists Z_{k+1})\ldots(\exists Z_n)\,\big[\\ \quad\quad F_2\,(X_{tid},Y_{h+1},\ldots,Y_k,Z_{k+1},\ldots,Z_n)\wedge\\ \quad\quad R\,(X_1,\ldots,X_h,D\,(X_{h+1},Y_{h+1}),\ldots,D\,(X_k,Y_k),Z_{k+1},\ldots,Z_n)\,\big]\big]\ ;\end{array}\tag{4}$$

the knowledge about r and f_1 deduced from the knowledge of f_2 is expressed by

$$\begin{array}{l}(\forall X_{tid})\,(\forall X_{h+1})\ldots(\forall X_k)\,(\forall X_{k+1})\ldots(\forall X_n)\,\big[\\ \quad F_2\,(X_{tid},X_{h+1},\ldots,X_k,X_{k+1},\ldots,X_n)\\ \quad\quad\Rightarrow\\ \quad(\exists Y_1)\ldots(\exists Y_h)\,(\exists Z_{h+1})\ldots(\exists Z_k)\,\big[\\ \quad\quad F_1\,(X_{tid},Y_1,\ldots,Y_h,Z_{h+1},\ldots,Z_k)\wedge\\ \quad\quad R\,(Y_1,\ldots,Y_h,D\,(Z_{h+1},X_{h+1}),\ldots,D\,(Z_k,X_k),X_{k+1},\ldots,X_n)\,\big]\big]\ ;\end{array}\tag{5}$$

the knowledge about f_1 and f_2 deduced from the knowledge of r as well as the knowledge about r deduced from f_1 and f_2 is expressed by

$$\begin{array}{l}(\forall X_1)\ldots(\forall X_h)\,(\forall X_{h+1})\ldots(\forall X_k)\,(\forall X_{k+1})\ldots(\forall X_n)\,\big[\\ \quad R\,(X_1,\ldots,X_h,X_{h+1},\ldots,X_k,X_{k+1},\ldots,X_n)\\ \quad\quad\Leftrightarrow\\ \quad(\exists Z_{tid})\,(\exists Y_{h+1})\ldots(\exists Y_k)\,\big[\\ \quad\quad F_2\,(Z_{tid},Y_{h+1},\ldots,Y_k,X_{k+1},\ldots,X_n)\wedge\\ \quad\quad F_1\,(Z_{tid},X_1,\ldots,X_h,E\,(X_{h+1},Y_{h+1}),\ldots,E\,(X_k,Y_k))\,\big]\big]\ ;\end{array}\tag{6}$$

the knowledge about inequalities in r based on f_1 is expressed by

$$\begin{array}{l}(\forall X_{tid})\,(\forall X'_{tid})\,(\forall X_1)\ldots(\forall X_h)\,(\forall X_{h+1})\ldots(\forall X_k)\,\left(\forall X'_{h+1}\right)\ldots(\forall X'_k)\,\big[\\ \quad\big[F_1\,(X_{tid},X_1,\ldots,X_h,X_{h+1},\ldots,X_k)\wedge\\ \quad\ F_1\left(X'_{tid},X_1,\ldots,X_h,X'_{h+1},\ldots,X'_k\right)\wedge(X_{tid}\not\equiv X'_{tid})\big]\\ \quad\quad\Rightarrow\\ \quad(\exists Y_{h+1})\ldots(\exists Y_n)\,(\exists Z_{h+1})\ldots(\exists Z_n)\,\big[\\ \quad\quad R\,(X_1,\ldots,X_h,Y_{h+1},\ldots,Y_k,Y_{k+1},\ldots,Y_n)\wedge\\ \quad\quad R\,(X_1,\ldots,X_h,Z_{h+1},\ldots,Z_k,Z_{k+1},\ldots,Z_n)\wedge\bigvee_{j=h+1}^{n}(Y_j\not\equiv Z_j)\,\big]\big]\ ;\end{array}\tag{7}$$

and the knowledge about inequalities in r based on f_2 is expressed by

$$\begin{array}{l}(\forall X_{tid})\,(\forall X'_{tid})\,(\forall X_{h+1})\ldots(\forall X_k)\,\left(\forall X'_{h+1}\right)\ldots(\forall X'_k)\,(\forall X_{k+1})\ldots(\forall X_n)\,\big[\\ \quad\big[F_2\,(X_{tid},X_{h+1},\ldots,X_k,X_{k+1},\ldots,X_n)\wedge\\ \quad\ F_2\left(X'_{tid},X'_{h+1},\ldots,X'_k,X_{k+1},\ldots,X_n\right)\wedge(X_{tid}\not\equiv X'_{tid})\big]\\ \quad\quad\Rightarrow\\ \quad(\exists Y_1)\ldots(\exists Y_k)\,(\exists Z_1)\ldots(\exists Z_k)\,\big[\\ \quad\quad R\,(Y_1,\ldots,Y_h,Y_{h+1},\ldots,Y_k,X_{k+1},\ldots,X_n)\wedge\\ \quad\quad R\,(Z_1,\ldots,Z_h,Z_{h+1},\ldots,Z_k,X_{k+1},\ldots,X_n)\wedge\bigvee_{j=1}^{k}(Y_j\not\equiv Z_j)\,\big]\big]\ .\end{array}\tag{8}$$

This view on r and f_2 is referred to as the set of sentences db_R containing the sentences (4), (5), (6), (7) and (8).

Strictly speaking, db_R alone does not provide *any* knowledge about the relational instance r; instead, only the combination of db_{f_1} and db_R describes the knowledge about r that is available to an attacker. The essential part of this insight is formally captured by the following proposition.

Proposition 2. *Under the assumptions of Def. 6, the sentences (4), (5), (6), (7) and (8) of db_R are satisfied by the DB-Interpretation $\mathcal{I}_r$, i.e., $\mathcal{I}_r \models_M db_R$.*

Proof. Omitted. See the informal explanations before Definition 6. □

Note that – in contrast to sentence (6) – the equivalence does *not* hold for the sentences (4) and (5), as it can be shown by a straightforward example.

Finally, we have to model the confidentiality policy logic-orientedly. A confidentiality constraint $c \subseteq A_R$ claims that each combination of (cleartext-)values allocated to the attributes of c should not be revealed to an attacker completely. To specify this semantics more precisely, it is assumed that c only protects those combinations of values which are explicitly allocated to the attributes of c in a tuple of r. In contrast, an attacker may get to know that a certain combination of values is *not* allocated to the attributes of c in any tuple of r.

The wish to protect a certain combination of values $(v_{i_1}, \ldots, v_{i_\ell}) \in Dom^{|c|}$ is modelled as a "potential secret" in the form of a sentence $(\exists \boldsymbol{X})\, R(t_1, \ldots, t_n)$ in which $t_j := v_j$ holds for each $j \in \{i_1, \ldots, i_\ell\}$ and all other terms are existentially quantified variables. To protect *each* of the infinitely many combinations, regardless of whether it is contained in a tuple of r or not, we use a single open formula with free variables $X_{i_1}, \ldots, X_{i_\ell}$ like an open query as follows.

Definition 7 (Confidentiality Policy). *Let $\mathcal{C}$ be a set of confidentiality constraints over schema $\langle R|A_R|SC_R\rangle$ according to Def. 2. Considering a confidentiality constraint $c_i \in \mathcal{C}$ with $c_i = \{a_{i_1}, \ldots, a_{i_\ell}\} \subseteq \{a_1, \ldots, a_n\} = A_R$ and the set $A_R \setminus c_i = \{a_{i_{\ell+1}}, \ldots, a_{i_n}\}$, constraint c_i is modelled as a* potential secret

$$\Psi_i(\boldsymbol{X_i}) \;:=\; (\exists X_{i_{\ell+1}}) \ldots (\exists X_{i_n})\, R(X_1, \ldots, X_n)\,,$$

which is a formula in the language $\mathscr{L}$. Thereby $\boldsymbol{X_i} = (X_{i_1}, \ldots, X_{i_\ell})$ is the vector of free variables contained in $\Psi_i(\boldsymbol{X_i})$. The set containing exactly one potential secret $\Psi_i(\boldsymbol{X_i})$ constructed as above for every confidentiality constraint $c_i \in \mathcal{C}$ is called potsec($\mathcal{C}$). Moreover, the expansion ex(*potsec*($\mathcal{C}$)) *contains all ground substitutions over Dom of all formulas in potsec($\mathcal{C}$).*

Example 7. For our example, $c_2 = \{\texttt{Name}, \texttt{Illness}\}$ is modelled as $\Psi_2(\boldsymbol{X_2}) := (\exists X_S)(\exists X_H)(\exists X_D)R(X_S, X_N, X_I, X_H, X_D)$ with free variables $\boldsymbol{X_2} = (X_N, X_I)$.

4 Inference-Proofness of Fragmentation

Until now the logic-oriented modelling of an attacker's view only comprises knowledge the attacker can deduce from the outsourced fragment instance f_1, which is supposed to be visible to him. Additionally, however, the attacker might also employ a priori knowledge to draw harmful inferences.

Example 8. As in Example 2, suppose the attacker knows that Doctor `White` is a psychiatrist only treating patients suffering from the `Borderline`-syndrome:

$$(\forall X_S)(\forall X_N)(\forall X_I)(\forall X_H)[R(X_S, X_N, X_I, X_H, \texttt{White}) \Rightarrow (X_I \equiv \texttt{Borderline})] \ .$$

This knowledge enables the attacker to conclude that patient `Hellmann` suffers from the illness `Borderline`-syndrome, thereby violating confidentiality constraint $c_2 = \{\texttt{Name}, \texttt{Illness}\}$. Moreover, let the attacker additionally know that all patients suffering from the `Borderline`-syndrome have hurt themselves:

$$(\forall X_S)(\forall X_N)(\forall X_H)(\forall X_D)[R(X_S, X_N, \texttt{Borderline}, X_H, X_D) \Rightarrow (X_N \equiv X_H)] \ .$$

The attacker can then draw the conclusion that patient `Hellmann` has been hurt by `Hellmann`, thereby violating $c_3 = \{\texttt{Name}, \texttt{HurtBy}\}$.

Following the framework of CIE [3], we aim at achieving a sophisticated kind of confidentiality taking care of an attacker's (postulated) a priori knowledge. This a priori knowledge is modelled as a finite set *prior* of sentences in $\mathscr{L}$ containing only R and $\equiv$ as predicate symbols. Moreover, we always assume that the semantic constraints SC_R declared in the relational schema are publicly known, i.e., $SC_R \subseteq \mathit{prior}$. Intuitively, we then would like to guarantee that a fragmentation is *inference-proof* in the sense that – from the attacker's point of view – each of the potential secrets might *not* be true in the original relational instance r. More formally: for each potential secret $\Psi_i(\boldsymbol{v_i}) \in \mathrm{ex}(\mathit{potsec}(\mathcal{C}))$ there should exist an alternative instance r' over $\langle R|A_R|SC_R\rangle$ that witnesses the non-entailment $db_{f_1} \cup db_R \cup \mathit{prior} \not\models_{DB} \Psi_i(\boldsymbol{v_i})$. Clearly, deciding on non-entailment, equivalently finding a suitable witness, is computationally infeasible in general. Accordingly, we will have to restrict on approximations and special cases.

Regarding approximations, we might straightforwardly require for the witness r' that for at least one $m \in \{i_1, \ldots, i_\ell\}$ the value v_m appearing in the potential secret must *not* occur under the attribute a_m. Accordingly, we could try to substitute v_m in the original instance r by a newly selected constant symbol v^* to obtain r'. However, we also have to preserve indistinguishability of r and r' by the attacker, and thus m has to be chosen such that $a_m \notin (A_{F_1} \setminus \mathcal{E})$. Furthermore, to fully achieve indistinguishability, the alternative instance r' has to coincide with the original instance r on the part visible in fragment f_1, i.e., $\mathcal{I}_{r'} \models_M db_{f_1}$, and modifying the original instance r into the alternative r' should preserve satisfaction of the a priori knowledge, i.e., $\mathcal{I}_{r'} \models_M \mathit{prior}$.

Regarding special cases, we will adapt two useful properties known from relational database theory [1]. *Genericity* of a sentence in $\mathscr{L}$ perceives constant symbols as being atomic and uninterpreted. Intuitively, all knowledge about a constant symbol arises from its occurrences in the relational instance r. Clearly, sentences with "essential" occurrences of constant symbols will not be generic. But in general "essential" occurrences of constant symbols are difficult to identify. Moreover, renaming v_m by v^* should not modify the fragment f_1 that is visible to the attacker. *Typedness* restricts the occurrences of a variable within a sentence to a single attribute (column), and thus prevents a "transfer of information" from a visible attribute to a hidden one.

We will now state our main result about the achievements of fragmentation with encryption regarding preservation of confidentiality against an attacker who only has access to one of the fragment instances, here exemplarily to fragment instance f_1. Facing the challenges discussed above, this main result exhibits a sufficient condition for confidentiality. An inference-proof fragmentation of the running example in terms of Theorem 1 is presented in Example 9 of Sect. 5.

Theorem 1 (Inference-Proofness on Schema Level). *Let $\langle R|A_R|SC_R\rangle$ be a relational schema with $A_R = \{a_1, \ldots, a_n\}$ and $(\mathcal{F}, \mathcal{E})$ be a fragmentation with fragment $\langle F_1|A_{F_1}|SC_{F_1}\rangle \in \mathcal{F}$ that is confidential w.r.t. a set $\mathcal{C}$ of confidentiality constraints. Moreover, let $SC_R \subseteq prior$ be a set of sentences in $\mathscr{L}$ containing only R and $\equiv$ as predicate symbols, satisfying the following restrictions:*

- Untyped dependencies with constants: *each $\Gamma \in prior$ is in the syntactic form of $(\forall \boldsymbol{x})(\exists \boldsymbol{y})[\bigvee_{j=1,\ldots,p} \neg A_j \vee A_{p+1}]$ with A_l being an atom of the form $R(t_{l,1}, \ldots, t_{l,n})$ or $(t_{p+1,1} \equiv t_{p+1,2})$ and $t_{j,i}$ is a variable or a constant symbol; moreover, w.l.o.g., equality predicates may only occur positively, and there might also be a conjunction of positively occurring R-atoms.*
- Satisfiability: *prior is DB-satisfiable and each $\Gamma \in prior$ is not DB-tautologic (and thus: each $\Gamma \in prior$ is range-restricted and does not contain an existentially quantified variable in the negated atoms (premises)).*
- Compatibility with $(\mathcal{F}, \mathcal{E})$ and $\mathcal{C}$: *there is a subset $M \subseteq \{h+1, \ldots, n\}$ s.t.*
  (1) $M \cap \{i_1, \ldots, i_\ell\} \neq \emptyset$ for each $c_i \in \mathcal{C}$ with $c_i = (a_{i_1}, \ldots, a_{i_\ell})$;
  (2) for each $\Gamma \in prior$ there exists a partitioning $\mathcal{X}_1^\Gamma \dot\cup \mathcal{X}_2^\Gamma = Var$ s.t.
    (i) for each atom $R(t_1, \ldots, t_n)$ of Γ
      - *for all $j \in \{1, \ldots, n\} \setminus M$ term t_j can either be a (quantified) variable of $\mathcal{X}_1^\Gamma$ or a constant symbol of Dom,*
      - *for all $j \in M$ term t_j must be a (quantified) variable of $\mathcal{X}_2^\Gamma$,*

    (ii) for each atom $(X_i \equiv X_j)$ of Γ either $X_i, X_j \in \mathcal{X}_1^\Gamma$ or $X_i, X_j \in \mathcal{X}_2^\Gamma$,
    (iii) for each atom $(X_i \equiv v)$ of Γ with $v \in Dom$ variable X_i is in $\mathcal{X}_1^\Gamma$.

Then, inference-proofness *is achieved: For each instance r over $\langle R|A_R|SC_R\rangle$ with fragment instance f_1 such that $\mathcal{I}_r \models_M prior$ and for each potential secret $\Psi_i(\boldsymbol{v_i}) \in \mathrm{ex}(potsec(\mathcal{C}))$ we have $db_{f_1} \cup db_R \cup prior \not\models_{DB} \Psi_i(\boldsymbol{v_i})$,*
i.e., there exists an alternative instance r' over $\langle R|A_R|SC_R\rangle$ s.t.

(a) $\mathcal{I}_{r'} \models_M db_{f_1} \cup db_R \cup prior$, and
(b) $\mathcal{I}_{r'} \not\models_M \Psi_i(\boldsymbol{v_i})$.

Proof (sketch). Consider any $\Psi_i(\boldsymbol{v_i}) \in \mathrm{ex}(potsec(\mathcal{C}))$ with $\boldsymbol{v_i} = (v_{i_1}, \ldots, v_{i_\ell})$. Then $c_i := \{a_{i_1}, \ldots, a_{i_\ell}\} \in \mathcal{C}$, and thus by the assumptions there is an attribute $a_m \in c_i$ with $m \in M$; moreover, either $a_m \in \mathcal{E}$ or $a_m \in (\bar{A}_{F_2} \setminus \mathcal{E})$.

Starting the construction of r' and thus of the induced $\mathcal{I}_{r'}$, to ensure $\mathcal{I}_{r'} \models_M db_{f_1}$ according to Proposition 1, we define $f_1' := f_1$ and $\mathcal{I}_{r'}(F_1) := f_1'$.

Continuing the construction of $\mathcal{I}_{r'}$, we select a constant symbol $v^* \neq v_m$ from the infinite set $\mathcal{U}$ that does not occur in the finite active domain of $\pi_M(r)$ and define a bijection $\varphi : \mathcal{U} \to \mathcal{U}$ such that $\varphi(v_m) = v^*$ and no value of $\pi_M(r)$ is mapped to v_m. Then we extend φ to a tuple transformation φ^* that maps

a value v for an attribute $a_j \in A_R$ with $j \in M$ to $\varphi(v)$ and each value for an attribute $a_j \in A_R$ with $j \notin M$ to itself, and define $r' := \varphi^*[r]$. Accordingly, the predicate symbol R is interpreted by $\mathcal{I}_{r'}(R) := r'$.

The instance r' and its fragment instance f_1' together uniquely determine the corresponding fragment instance f_2' – whose constructability is guaranteed by the group properties of *Enc* – and thus we define $\mathcal{I}_{r'}(F_2) := f_2'$.

By the selection of v^* and the definition of φ, we immediately have $\mathcal{I}_{r'} \not\models_M \Psi_i(\boldsymbol{v_i})$, and thus $\mathcal{I}_{r'}$ complies with property (b). Furthermore, by the construction and according to Proposition 2, $\mathcal{I}_{r'} \models_M db_R$. Finally, we outline the argument to verify the remaining part of property (a), namely $\mathcal{I}_{r'} \models_M \mathit{prior}$.

We consider the following $\Gamma \in \mathit{prior}$ (other cases are treated similarly):

$$(\forall \boldsymbol{x})(\exists \boldsymbol{y})[\bigvee_{j=1,\ldots,p} \neg R(t_{j,1},\ldots,t_{j,n}) \vee R(t_{p+1,1},\ldots,t_{p+1,n})]\,,$$

where $\{t_{j,1},\ldots,t_{j,n}\} \subseteq \boldsymbol{x} \cup \mathit{Dom}$ for $j \in \{1,\ldots,p\}$ and $\{t_{p+1,1},\ldots,t_{p+1,n}\} \subseteq \boldsymbol{x} \cup \boldsymbol{y} \cup \mathit{Dom}$. To demonstrate $\mathcal{I}_{r'} \models_M \Gamma$, we inspect any variable substitution $\sigma' : \boldsymbol{x} \to \mathit{Dom}$. If there exists $j \in \{1,\ldots,p\}$ such that $\mathcal{I}_{r'}^{\sigma'} \models_M \neg R(t_{j,1},\ldots,t_{j,n})$, we are done.

Otherwise, for all $j \in \{1,\ldots,p\}$ we have $\mathcal{I}_{r'}^{\sigma'} \not\models_M \neg R(t_{j,1},\ldots,t_{j,n})$ and thus for each tuple $\mu_j' := (\sigma'(t_{j,1}),\ldots,\sigma'(t_{j,n}))$ we have $\mu_j' \in r'$. Since $r' := \varphi^*[r]$, for all $j \in \{1,\ldots,p\}$ there exists $\mu_j \in r$ such that $\varphi^*[\mu_j] = \mu_j'$. Now exploiting the properties of the set M – essentially, for each term exactly one case of the definition of φ^* applies – we can construct a variable substitution $\sigma : \boldsymbol{x} \to \mathit{Dom}$ such that $\mu_j = (\sigma(t_{j,1}),\ldots,\sigma(t_{j,n}))$ and, accordingly, $\mathcal{I}_r^{\sigma} \not\models_M \neg R(t_{j,1},\ldots,t_{j,n})$.

Since $\mathcal{I}_r \models_M \Gamma$, there exists a variable substitution $\tau : \boldsymbol{y} \to \mathit{Dom}$ such that $\mathcal{I}_r^{\sigma|\tau} \models_M R(t_{p+1,1},\ldots,t_{p+1,n})$, i.e., $\mu_{p+1} := (\sigma|\tau(t_{p+1,1}),\ldots,\sigma|\tau(t_{p+1,n})) \in r$. By the definition of r', we have $\mu_{p+1}' := \varphi^*[\mu_{p+1}] \in r'$.

Exploiting the properties of M and using τ, we can construct a variable substitution $\tau' : \boldsymbol{y} \to \mathit{Dom}$ such that $\mu_{p+1}' = (\sigma'|\tau'(t_{p+1,1}),\ldots,\sigma'|\tau'(t_{p+1,n}))$. Hence, $\mathcal{I}_{r'}^{\sigma'|\tau'} \models_M R(t_{p+1,1},\ldots,t_{p+1,n})$ and thus $\mathcal{I}_{r'}^{\sigma'|\tau'} \models_M \Gamma$. □

Theorem 1 provides a sufficient condition for inference-proofness on schema level, i.e., for *each* relational instance satisfying the a priori knowledge *prior*. In some situations, however, a security officer might aim at only achieving inference-proofness of a fixed particular relational instance r. Such a situation could be captured by a corollary. Essentially, if we know r and thus also f_1 in advance, we can inspect the usefulness of each implicational sentence $\Gamma \in \mathit{prior}$ of form $(\forall \boldsymbol{x})(\exists \boldsymbol{y})[\bigvee_{j=1,\ldots,p} \neg A_j \vee A_{p+1}]$ to derive harmful information for the specific situation. If r already satisfies $(\forall \boldsymbol{x})[\bigvee_{j=1,\ldots,p} \neg A_j]$, then we can completely discard Γ from the considerations. More generally, we could only consider the effects of Γ for those variable substitutions σ of $\boldsymbol{x}$ that make $[\bigvee_{j=1,\ldots,p} \neg A_j]$ *false* for r.

5 Creation of an Appropriate Fragmentation

If an attacker is supposed to have a priori knowledge, a fragmentation has to comply with this knowledge to guarantee inference-proofness in terms of Theorem 1.

F_1	tid	SSN	Illness	HurtBy	Doctor
	1	e_S^1	Borderline	e_H^1	White
	2	e_S^2	Laceration	e_H^2	Warren
	3	e_S^3	Laceration	e_H^3	Warren
	4	e_S^4	Concussion	e_H^4	Warren

F_2	tid	SSN	HurtBy	Name
	1	κ_S^1	κ_H^1	Hellmann
	2	κ_S^2	κ_H^2	Dooley
	3	κ_S^3	κ_H^3	McKinley
	4	κ_S^4	κ_H^4	McKinley

Fig. 3. Inference-proof fragmentation w.r.t. a priori knowledge of Example 8

Hence, an algorithm computing a fragmentation should not only determine an arbitrary fragmentation being confidential in terms of Def. 3. The algorithm should rather consider *all* of these fragmentations and select one complying with the user's a priori knowledge (if such a fragmentation exists).

Example 9. Reconsidering the a priori knowledge presented in Example 8, this knowledge does *not* compromise confidentiality if the fragmentation known from Fig. 1 is modified as depicted in Fig. 3. In terms of Theorem 1, for an attacker knowing f_1 the set M can be chosen to contain the indices of `SSN`, `HurtBy` and `Name` and for both sentences Γ_1 and Γ_2 of Example 8 the set of variables can be partitioned s.t., for both $i \in \{1,2\}$, $X_I, X_D \in \mathcal{X}_1^{\Gamma_i}$ and $X_S, X_N, X_H \in \mathcal{X}_2^{\Gamma_i}$.

In the following, an Integer Linear Program (ILP) (see [11]) computing a confidential fragmentation complying with an attacker's a priori knowledge is developed to solve this problem with the help of generic algorithms solving ILPs.[2] As the optimization goal the set of "encrypted attributes" is chosen to be minimized to reduce the costs for processing queries over the fragmented database as proposed in [2,9]. Other optimization goals are conceivable, too.

Given the attribute set A_R of an original schema $\langle R|A_R|SC_R\rangle$, a set $\mathcal{C}$ of confidentiality constraints and a set *prior* in terms of Theorem 1, the ILP presented in the following computes the attribute sets $\bar{A}_{F_1}$ and $\bar{A}_{F_2}$ as well as the set $\mathcal{E}$ of "encrypted attributes" of a fragmentation being confidential w.r.t to $\mathcal{C}$ and complying with *prior*. The ILP contains the following binary decision variables:

- A variable a_j^i, for both $i \in \{1,2\}$ and for each $a_j \in A_R$. If $a_j^i = 1$, attribute $a_j \in A_R$ is in $\bar{A}_{F_i}$; if $a_j^i = 0$, attribute $a_j \in A_R$ is *not* in $\bar{A}_{F_i}$.
- A variable a_j^e for each $a_j \in A_R$. If $a_j^e = 1$, attribute $a_j \in A_R$ is an "encrypted attribute"; if $a_j^e = 0$, attribute $a_j \in A_R$ is a "cleartext attribute".
- A variable m_j for each $a_j \in A_R$. If $m_j = 1$, the index of attribute a_j is in M; if $m_j = 0$, the index of attribute a_j is *not* in M.
- A variable X^Γ for each variable X contained in a sentence $\Gamma \in$ *prior*. If $X^\Gamma = 1$, variable X is in $\mathcal{X}_1^\Gamma$; if $X^\Gamma = 0$, variable X is in $\mathcal{X}_2^\Gamma$.

For each $\Gamma \in$ *prior* the set Var_j^Γ is assumed to contain X^Γ if Γ is built over an atom $R(t_1, \ldots, t_n)$ with t_j being the variable X (note that each variable might occur in different columns). Moreover, the set $const(\Gamma)$ is assumed to contain the index j, if Γ is built over an atom $R(t_1, \ldots, t_n)$ with t_j being a constant. Then, the ILP computing an appropriate fragmentation is defined as follows:

[2] For our prototype implementation "lp_solve" turned out to be an appropriate and fast ILP solver (see http://lpsolve.sourceforge.net/).

Minimize the number of "encrypted attributes", i.e., min: $\sum_{j=1}^{n} a_j^e$ s.t. the following constraints are fulfilled:

- "Cleartext attributes" in exactly one fragment, "encrypted ones" in both:
  $a_j^1 + a_j^2 = 1 + a_j^e$ for each $a_j \in A_R$
- For $i \in \{1, 2\}$, fragment $\langle F_i | A_{F_i} | SC_{F_i} \rangle$ fulfills all confidentiality constraints:
  $\sum_{a_j \in c} a_j^i \leq |c| - 1 + \sum_{a_j \in c} a_j^e$ for each $c \in \mathcal{C}$ and each $i \in \{1, 2\}$
- $M \subseteq \{h+1, \ldots, n\}$, i.e., M is a subset of attributes in A_{F_2}:
  $m_j \leq a_j^2$ for each $a_j \in A_R$
- M overlaps with the indices of the attributes of each $c \in \mathcal{C}$:
  $\sum_{a_j \in c} m_j \geq 1$ for each $c \in \mathcal{C}$
- For each formula $\Gamma \in \mathit{prior}$:
  - In each $R(t_1, \ldots, t_n)$ of Γ: for each t_j being a constant with $j \notin M$:
    $m_j = 0$ for each $j \in \mathit{const}(\Gamma)$
  - Partitioning of variables into $\mathcal{X}_1^\Gamma$ and $\mathcal{X}_2^\Gamma$:
    $X^\Gamma = 1 - m_j$ for $j \in \{1, \ldots, n\}$ with $\mathit{Var}_j^\Gamma \neq \emptyset$ and each $X^\Gamma \in \mathit{Var}_j^\Gamma$
  - In each atom $(X_i \equiv X_j)$: variables X_i, X_j belong to the same partition:
    $X_i^\Gamma = X_j^\Gamma$ for each atom $(X_i \equiv X_j)$
  - In each atom $(X \equiv v)$: variable X belongs to partition $\mathcal{X}_1^\Gamma$:
    $X^\Gamma = 1$ for each atom $(X \equiv v)$
- Each decision variable of this ILP is binary:
  $0 \leq x \leq 1$ for each integer decision variable x of this ILP

If the ILP solver outputs a feasible solution, an inference-proof fragmentation can be determined by constructing the sets $\bar{A}_{F_1}$, $\bar{A}_{F_2}$ and $\mathcal{E}$ of Def. 1 according to the allocation of the corresponding decision variables of the ILP.

Note that availability requirements such as storing a particular subset of attributes within the same (or even a particular) fragment or keeping the values of a particular attribute as cleartext values can be simply modelled by adding appropriate constraints, i.e., (in-)equations, to the ILP.

6 Conclusion and Future Work

Motivated by the question, whether splitting of data vertically over two semi-honest servers guarantees confidentiality, the fragmentation model introduced in [2,9] is formalized, then modelled logic-orientedly and subsequently analyzed w.r.t. its inference-proofness. This analysis considers an attacker employing his a priori knowledge to draw harmful inferences and provides a sufficient condition to decide whether a given combination of a fragmentation and a priori knowledge is inference-proof w.r.t. a given confidentiality policy. Additionally, a generic ILP formulation computing such an inference-proof fragmentation is developed.

As Theorem 1 only states a sufficient condition for inference-proofness, there might be a more relaxed, most desirably even necessary definition of a priori knowledge still guaranteeing inference-proofness. A full characterization of inference-proofness could also provide a basis for deciding on the existence of a secure fragmentation for a given setting.

Theorem 1 might be also enhanced in the spirit of k-anonymity by a more sophisticated definition of confidentiality guaranteeing that an "invisible value" cannot be narrowed down to a set of possible values of a certain cardinality. A further analysis of confidentiality assuming that commonly used encryption functions such as AES or RSA (which do *not* satisfy the group properties) come into operation is desirable, too. Although a formal analysis based on probability theory and complexity theory is indispensable to guarantee profound statements, we expect these encryption functions to be "sufficiently secure" in practice.

In this article and previously in [5] each one of two existing approaches to achieve confidentiality by vertical fragmentation is analyzed. As a third approach – using an arbitrary number of fragments which are *all* supposed to be known to an attacker – is presented in [8], a formal analysis of this approach in the spirit of Theorem 1 might be another challenging task for future work.

References

1. Abiteboul, S., Hull, R., Vianu, V.: Foundations of Databases. Addison-Wesley, Reading (1995)
2. Aggarwal, G., Bawa, M., Ganesan, P., Garcia-Molina, H., Kenthapadi, K., Motwani, R., Srivastava, U., Thomas, D., Xu, Y.: Two can keep a secret: A distributed architecture for secure database services. In: CIDR 2005. pp. 186–199 (2005)
3. Biskup, J.: Inference-usability confinement by maintaining inference-proof views of an information system. International Journal of Computational Science and Engineering 7(1), 17–37 (2012)
4. Biskup, J., Bonatti, P.A.: Controlled query evaluation with open queries for a decidable relational submodel. Annals of Mathematics and Artificial Intelligence 50(1–2), 39–77 (2007)
5. Biskup, J., Preuß, M., Wiese, L.: On the Inference-Proofness of Database Fragmentation Satisfying Confidentiality Constraints. In: Lai, X., Zhou, J., Li, H. (eds.) ISC 2011. LNCS, vol. 7001, pp. 246–261. Springer, Heidelberg (2011)
6. Biskup, J., Wiese, L.: A sound and complete model-generation procedure for consistent and confidentiality-preserving databases. Theoretical Computer Science 412(31), 4044–4072 (2011)
7. Ciriani, V., De Capitani di Vimercati, S., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P.: Keep a few: Outsourcing data while maintaining confidentiality. In: Backes, M., Ning, P. (eds.) ESORICS 2009. LNCS, vol. 5789, pp. 440–455. Springer, Heidelberg (2009)
8. Ciriani, V., De Capitani di Vimercati, S., Foresti, S., Jajodia, S., Paraboschi, S., Samarati, P.: Combining fragmentation and encryption to protect privacy in data storage. ACM Transactions on Information and System Security 13(3) (2010)
9. Ganapathy, V., Thomas, D., Feder, T., Garcia-Molina, H., Motwani, R.: Distributing data for secure database services. Transactions on Data Privacy 5(1), 253–272 (2012)
10. Hacigümüs, H., Mehrotra, S., Iyer, B.R.: Providing database as a service. In: ICDE 2002. pp. 29–40. IEEE Computer Society, Los Alamitos (2002)
11. Korte, B., Vygen, J.: Combinatorial Optimization: Theory and Algorithms. Algorithms and Combinatorics, Springer, Heidelberg, 5th edn. (2012)