Inference-Proof Materialized Views

Extended Abstract of PhD Thesis

Marcel Preuß

1 Motivation and Goals

Nowadays, data publishing is ubiquitous: governments provide data about matters of public concern, companies release project-related data to partners and people disclose data about their private lives. But usually only certain portions of some data are appropriate to be shared, as data often contains sensitive information, and even an authorized receiver of some data might act maliciously as an *adversary* and thereby try to gain confidential knowledge by (actively) uncovering some pieces of information *not* intended for him. Hence, beside the sharing of data, another key goal of data publishing lies in the preservation of confidentiality requirements. This applies in particular to data containing personal information.

Motivated by the well-known approaches of k-anonymization and ℓ-diversification, which aim at achieving confidentiality by generalizing (and thereby weakening) some values of a given dataset to wider sets of possible values, this thesis proposes a novel approach to weaken an original relational database instance, which is supposed to be complete, to a materialized *weakened view* on this instance. Such a weakened view can be securely published instead of the original database instance, as it is *inference-proof* in the spirit of the framework of "Controlled Interaction Execution" and does hence provably *not* enable an adversary to compromise a *confidentiality policy* – even if this adversary tries to (logically) deduce confidential knowledge on the basis of

- a weakened view released to him,
- his general awareness of the protection mechanism,
- some a priori knowledge he might possibly have about the original database instance or the world in general and
- the assumed completeness of database instances, allowing for reasoning about (negative) knowledge *not* valid within an original database instance.

To achieve inference-proofness one must hence protect the knowledge embodied in (elements of) a confidentiality policy by suitably confining an adversary's possibilities of gaining information to such an extent that this adversary is provably *not* able to infer a piece of confidential knowledge by employing his reasoning capabilities. However, this presupposes that data is handled on the level of its semantics. As a consequence, inference-proofness can usually *not* be achieved with the help of approaches traditionally used to limit access to sensitive data, as these traditional approaches essentially operate on the level of raw data and do *not* take the semantics of this data into account.

2 Conceptual Approach

In this thesis all knowledge (embodied in the considered data) is modeled within a *first-order logic* framework, as it is generally well understood that first-order logic provides a solid foundation for relational databases and as the semantics of first-order logic is moreover comprehensive enough to capture an adversary's reasoning capabilities by means of *implications* (or entailments) between first-order logic sentences. But implication between first-order logic sentences is well-known to be computationally undecidable in general and still computationally hard for expressive and decidable subclasses of first-order logic. For the construction of a weakening approach, which is *efficient* enough to handle even large input instances, this thesis consequently resorts to a less expressive (but still semantically useful) subclass of first-order logic for the modeling of confidentiality policies, which is referred to as "existentially quantified atoms" and allows to reduce the implication problem to an easy to solve pattern matching problem.

2.1 Construction of Weakened Views

To actually construct inference-proof weakened views within a first-order logic modeling, all original database tuples compromising a confidentiality policy are (whenever possible) replaced by weaker but true disjunctions (preferably) consisting of elements of the considered confidentiality policy. Although this disjunctive knowledge deliberately introduces uncertainty about confidential knowledge, it still provides more information about the original database instance than complete refusals of confidential knowledge and thus significantly improves availability. A resulting weakened view is then constructed on the basis of three (semantically disjoint) classes of knowledge:

- the *positive knowledge* consisting of the definitely harmless database tuples,
- the *disjunctive knowledge* consisting of the constructed disjunctions and
- the *negative knowledge* consisting of all knowledge, which is
  - neither part of the positive knowledge
  - nor captured within a disjunct of the disjunctive knowledge.

At first leaving an adversary's a priori knowledge aside, a *generic* weakening approach is developed, which allows for the construction of disjunctions of any length ≥ 2 to weaken an adversary's possible gain of confidential knowledge. Thereby, the achieved level of confidentiality varies with the length of weakening disjunctions, as longer disjunctions of policy elements obviously provide more alternatives which (combination of) policy elements of such a disjunction might be satisfied by a considered original database instance. More precisely, a weakening disjunction of length k does provably *not* enable an adversary to distinguish whether one specific alternative provided by this disjunction or (at least) one of the $k-1$ other alternatives provided by this disjunction is satisfied by the original database instance.

The basic technique of grouping elements of a confidentiality policy to weakening disjunctions naturally raises the question which (subset of) policy elements should be grouped together to a weakening disjunction: in terms of confidentiality all alternatives provided by a weakening disjunction should be equally probable to prevent an adversary from excluding certain alternatives from being true, and in terms of availability such a disjunction should provide as much useful information as possible. Hence, an *admissibility criterion* specifying which policy elements might be possibly grouped together to an admissible disjunction needs to be specified. But as such a notion of a semantically meaningful grouping of policy elements of course highly depends on the considered application scenario, this criterion is deliberately left generic to keep the weakening approach applicable for different application scenarios.

Although the generic weakening approach fully specifies the construction of inference-proof weakened views on the declarative level, it does *not* provide an algorithmic instantiation for its subroutine clustering (i.e., algorithmically grouping) the elements of a confidentiality policy to weakening disjunctions – thereby employing only a minimum number of (artificial) additional policy elements, whose construction may be necessary to find a feasible clustering. As this problem is formally proved to be NP-hard for the construction of disjunctions of a minimum length of 3, an efficient algorithmic solution to this clustering problem can (as long as $\mathsf{NP} \neq \mathsf{P}$ is supposed to hold) only be found for the construction of weakening disjunctions of length 2.

Such an algorithmic solution can actually be implemented on the basis of well-known and efficient algorithms for the computation of *maximum matchings* on general (i.e., *not* necessarily bipartite) graphs modeling each policy element to be clustered as a vertex and each admissible disjunction of length 2, which is feasible according to the above mentioned admissibility criterion, as an edge. As disjunctions of length 2 are the shortest possible non-trivial disjunctions, they only provide a minimum number of alternatives for each policy element to be protected and consequently lead to an *availability-maximizing* instantiation of the generic weakening approach.

2.2 Introducing A Priori Knowledge

This availability-maximizing instantiation is furthermore extended to be also confidentiality preserving under scenarios, in which an adversary has some *a priori knowledge*.

This a priori knowledge is supposed to be expressible within a suitably restricted subclass of so-called "Tuple Generating Dependencies", which are well-known semantic constraints in the field of relational databases.

In scenarios without a priori knowledge the inference-proofness of a weakened view crucially relies on a *strict isolation* of its disjunctive knowledge – which aims at *not* revealing the real truth values of the elements of its weakening disjunctions – from the definite knowledge an adversary can gain about the original database instance. This isolation is achieved by the construction of weakened views, guaranteeing that there is *no* implication relationship between a disjunct of the disjunctive knowledge and the positive knowledge or the negative knowledge of a weakened view. Even within the disjunctive knowledge care is taken that there is *no* implication relationship between each pair of different disjuncts. This isolation follows the goal that the satisfaction or non-satisfaction of one element of a weakening disjunction can *not* be concluded on the basis of the satisfaction or non-satisfaction of another piece of knowledge the adversary is aware of.

But existing a priori knowledge of the considered subclass of tuple generating dependencies can break this isolation up, as (transitive chains of) these dependencies might enable an adversary to conclude

- the existence of a certain (possibly confidential) database tuple on the basis of another existing database tuple by so-called *forward chaining* or
- the non-existence of (a set of) certain database tuples (thereby possibly eliminating alternatives provided by weakening disjunctions) on the basis of another non-existing database tuple by so-called *backward chaining.*

To reestablish a sufficient level of isolation, a considered confidentiality policy is first extended by further policy elements to disable the above mentioned forward and backward chaining with the help of those dependencies *interfering* with the confidentiality policy in a possibly harmful way. As such a policy extension can of course lead to new interferences between the a priori knowledge and the (partly extended) confidentiality policy, a confidentiality policy must be extended iteratively until a fixpoint is reached.

Moreover, weakening disjunctions (used to enforce an extended confidentiality policy) should *not* exclusively provide alternatives, each of which allows to imply the same piece of confidential knowledge with the help of (transitive chains of) dependencies. To create a basis for an efficient detection of such weakening disjunctions and to thereby avoid their construction, an adversary's a priori knowledge is partitioned such that all (transitive chains of) dependencies, which might enable the implication of the same piece of confidential knowledge, are guaranteed to be in the same partition. This finally leads to an extended weakening approach, whose inference-proofness is formally verified under the considered a priori knowledge, but still remains computationally efficient.

Unfortunately, it turns out that some elements of a confidentiality policy might *not* be protected sufficiently by weakening disjunctions within scenarios considering an adversary having a priori knowledge. The extended weakening approach then needs to *refuse* these pieces of knowledge completely to guarantee a proper enforcement of the confidentiality policy. This class of refused knowledge can be seen as a special subclass of disjunctive knowledge, as refusals can be expressed as tautological disjunctions.

3 Experimental Evaluation

To demonstrate the *practical efficiency* of the extended availability-maximizing weakening algorithm, a prototype implementation has been developed and evaluated under different experiment setups. These experiment setups systematically vary those parameters for an essentially random generation of input instances, which have a crucial impact on the runtime of the prototype implementation. Roughly summarized, these experiments demonstrate that the prototype implementation is able to process input instances in the order of magnitude of

- database instances with 1 000 000 tuples,
- confidentiality policies with up to 100 000 elements and
- an adversary's a priori knowledge with up to 2500 dependencies

in about one minute of time on a machine with 16 native CPU cores, each of which can logically handle two threads simultaneously due to hyperthreading. As parallelization turns out to scale nearly optimally for the most time-consuming subroutines of the prototype implementation, more powerful machines might lead to even better results.

For some restricted application scenarios, in which the (non-parallelized) computation of a maximum matching – to cluster policy elements to weakening disjunctions – turns out to have by far the greatest impact on the overall runtime, a slightly modified weakening algorithm employing a *heuristic* for the construction of (almost) maximum matchings is presented. This modification leads to a considerable speedup and results only in a minor loss of availability due to a slightly increased number of artificial additional policy elements occurring in the resulting weakening disjunctions.

For a prototype implementation the weakening approach of course needs to be instantiated with a concrete admissibility criterion – specifying which policy elements might be possibly grouped together to an admissible disjunction – to be fully specified. Therefore, an admissibility criterion called "interchangeability" is developed, which (locally) maximizes availability within a disjunction of length 2 in the sense that both of its disjuncts differ in only one constant parameter and thereby generalizes this constant parameter to a wider set of possible values.