M-PRIVACY FOR COLLABORATIVE DATA PUBLISHING

[1]V.Sakthivel, [2]G.Gokulakrishnan
[1]Pg Scholar, Department of Information Technology, Jayam College of Engineering and Technology, Dharmapuri
[2]Associate Professor/IT, Department of Information Technology, Jayam College of Engineering and Technology, Dharmapuri
[1]sakthivelvairam@gmail.com

Abstract: We are taking out the concerted data and data publishing issues for anonymizing detachment. Here data's are considered into two types of detachment, one is horizontally another one is vertically. Here the anonymizing data's are horizontally detachment at no of data providers. We notice the internally attacked by data providers. They are using its own records to conclude through by other third parties. We give the m-privacy condition. This condition is taking cover and satisfying the privacy rules. Then we have given the heuristic algorithms. This algorithm is using the no of corresponding groups of privacy rules. And recently using the adaptive ordering techniques for professionally inspected the m-privacy records. The above conditions and algorithms are using the data's are highly protected and safe with effectively.

1. INTRODUCTION

Privacy preserving data analysis and data publishing has received considerable attention in recent years as promising approaches for sharing data while preserving individual privacy. When the data are distributed among multiple data providers or data owners, two main settings are used for anonymization. One approach is for each provider to anonymize the data independently (anonymize-and-aggregate, which results in potential loss of integrated data utility. A more desirable approach is concerted data publishing which anonymizes data from all providers as if they would come from one source (aggregateand-anonymize), using either a trusted third-party (TTP) or Secure Multi-party Computation (SMC) protocols to do computations.

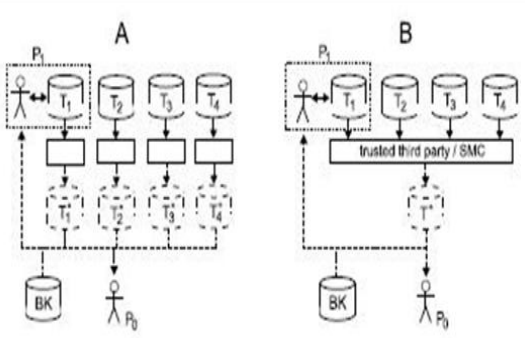


Figure 1: Distributed data publishing settings

Our goal is to publish an anonymized view of the integrated data such that a data recipient including the data providers will not be able to compromise the privacy of the individual records provided by other parties. Considering different types of malicious users and information they can use in attacks, we identify three main categories of attack scenarios. While the first two are addressed in existing work, the last one receives little attention and will be the focus of this paper.

A data recipient, e.g. P0, could be an attacker and attempts to infer additional information about the records using the published data (T*) and some background knowledge (BK) such as publicly available external data. Most literature on privacy preserving data publishing in a single provider setting considers only such attacks. Many of them adopt a weak or relaxed adversarial or Bayes-optimal privacy notion to protect against specific types of attacks by assuming limited background knowledge. For example, k-anonymity prevents identity disclosure attacks by requiring each equivalence group, records with the same quasi-identifier values, to contain at least k records. Representative constraints that prevent attribute disclosure attacks include l-diversity, which requires each equivalence group to contain at least l "well-represented" sensitive values, and t-closeness, which requires the distribution of a sensitive attribute in any equivalence group to be close to its distribution in the whole population.

They can attempt to infer additional information about data coming from other providers by analyzing the data received during the anonymization. A trusted third party (TTP) or Secure Multi-Party Computation (SMC) protocols can be used to guarantee there is no disclosure of intermediate information during the

anonymization. However, either TTP or SMC do not protect against data providers to infer additional information about other records using the anonymized data and their own data (discussed below). Since the problem is orthogonal to whether a TTP or SMC is used for implementing the algorithm, without loss of generality, we have assumed that all providers use a TTP for anonymization and note that an SMC variant can be implemented.

We define and address this new type of "insider attack" by data providers in this paper. In general, we define an m adversary as a coalition of m colluding data providers or data owners, who have access to their own data records as well as publicly available background knowledge BK and attempts to infer data records contributed by other data providers. Note that 0-adversary can be used to model the external data recipient, who has only access to the external background knowledge. Since each provider holds a subset of the overall data, this inherent data knowledge has to be explicitly modelled and checked when the data are anonymized using a weak privacy constraint and assuming no instance level knowledge.

We illustrate the m-adversary threats with an example shown in Table I. Assume that hospitals P1, P2, P3, and P4 wish to collaboratively anonymize their respective patient databases T1, T2, T3, and T4. In each database, Name is an identifier, f Age, Zip g is a quasi-identifier (QI), and Disease is a sensitive attribute. Ta* is one possible QI-group-based anonymization using existing approaches that guarantees k anonymity and l-diversity (k = 3, l = 2). Note that l diversity holds if each equivalence group contains records with at least l different sensitive attribute values. However, an attacker from the hospital P1, who has access to T1, may remove all records from Ta* that are also in T1 and find out that there is only one patient between 20 and 30 years old. Combining this information with background knowledge BK, P1 can identify Sara's record (highlighted in the table) and her disease Epilepsy. In general, multiple providers may collude with each other, hence having access to the union of their data, or a user may have access to multiple databases, e.g. a physician switching to another hospital, and use the increased data knowledge to infer data at other nodes.

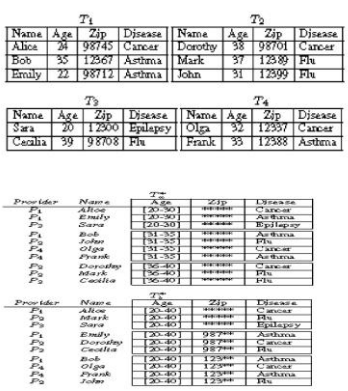


Figure 2: m-adversary and m-privacy example

Contributions

In this paper, we address the new threat by m-adversaries and make several important contributions. First, we introduce the notion of m-privacy that explicitly models the inherent data knowledge of an m-adversary and protects anonymized data against such adversaries with respect to a given privacy constraint. For example, an anonymization satisfies m-privacy with respect to l-diversity if the records in each equivalence group excluding ones from any m-adversary still satisfy l-diversity. In our example in Table I, Tb* is an anonymization that satisfies m-privacy (m = 1) with respect to k-anonymity and l-diversity (k = 3, l = 2).

Second, to address the challenges of checking a combinatorial number of potential m-adversaries, we present heuristic algorithms for efficiently verifying m-privacy given a set of Records. Our approach utilizes effective pruning strategies exploiting the equivalence group monotonicity property of privacy constraints and adaptive ordering techniques based on a novel notion of privacy fitness. Finally, we present a data provider-aware anonymization algorithm with adaptive strategies of checking m-privacy to ensure high utility and m privacy of sanitized data with efficiency. We experimentally show the feasibility and benefits of our approach using real world dataset.

2. M-PRIVACY DEFINITION

We first formally describe our problem setting. Then we present our m-privacy definition with respect to a given privacy constraint to prevent inference attacks by m-adversary, followed by its properties. Let T = t1, t2 . . . be a set of records horizontally distributed among n data providers P = P1, P2. . . Pn, such that Ti _ T is a set of records provided by Pi. We assume AS is a sensitive attribute with domain DS. If the records contain

multiple sensitive attributes then a new sensitive attribute AS can be defined as a Cartesian product of all sensitive attributes. Our goal is to publish an anonymized table T* while preventing any m-adversary from inferring AS for any single record.

2.1 m-Privacy

To protect data from external recipients with certain background knowledge BK, we assume a given privacy requirement C, defined by a conjunction of privacy constraints: C1 ^ C2 ^ . . . ^ Cw. If a set of records T* satisfies C, we say C (T*) true. Any of the existing privacy principles can be used as a component constraint. In our example (Table I), the privacy constraint C is defined as C = C1 ^ C2, where C1 is k-anonymity with k = 3, and C2 is l-diversity with l = 2. Both anonymized tables, Ta* and Tb* satisfies C, although as we have shown earlier, Ta* may be compromised by an m-adversary such as P1.

We now formally define a notion of m-privacy with respect, to a privacy constraint C, to protect the anonymized data against m-adversaries in addition to the external data recipients. The notion explicitly models the inherent data knowledge of an m-adversary, the data records they jointly contribute, and requires that each equivalence group, excluding any of those records owned by an m-adversary, still satisfies C. Definition 2.1: (m-PRIVACY) Given n data providers, a set of records T, and an anonymization mechanism A, an m-adversary I (m<=n-1) is a coalition of m providers, which jointly contributes a set of records TI. Sanitized records T* = A (T) satisfy m-privacy, i.e. are m-private, with respect to a privacy constraint C, if and only if, provider. Thus, each data provider may be able to breach privacy of records provided by others. In our example from Table I, Ta* satisfies only 0-privacy w.r.t. C = k-anonymity ^ l-diversity (k = 3, l = 2), while Tb* satisfies 1-privacy w.r.t.

The same C. m-Privacy is defined w.r.t. a privacy constraint C, and hence will inherit strengths and weaknesses of C. For example, if C is defined by k-anonymity, then ensuring m-privacy w.r.t. C will not protect against homogeneity attacker de Finetti attack. However, m-privacy w.r.t. C will protect against a privacy attack issued by any m-adversary, if and only if, C protects against the same privacy attack by any external data recipient. M-Privacy constraint is orthogonal to the privacy constraint C being used.

2.2 M-Privacy and Differential Privacy

Differential privacy does not assume specific background knowledge and guarantees privacy even if an attacker knows all records except the victim record. Thus, any statistical data (or records synthesized from the statistical data) satisfying differential privacy also satisfies (n-1)-privacy, i.e. maximum level of m-privacy, when any (n-1) providers can collude. While m-privacy w.r.t. any weak privacy notion does not guarantee unconditional privacy, it offers a practical trade off between preventing m-adversary attacks with bounded power m and the ability to publish generalized but truthful data records. In the rest of the paper, we will focus on checking and achieving m-privacy w.r.t. weak privacy constraints.

3. Monotonicity of Privacy Constraints

Generalization based monotonicity has been defined for privacy constraints in the literature (Definition 2.2) and has been used for designing efficient generalization algorithms to satisfy a privacy constraint. In this paper we will refer to it as generalization monotonicity. Definition 2.2: Generalization Monotonicity of a Privacy Constraint A privacy constraint C is generalization monotonic if and only if for any set of anonymized records T* satisfying C, all its further generalizations satisfy C as well. Generalization monotonicity assumes that original records T have been already anonymized and uses them for further generalizations. In this paper, we also introduce more general, record-based definition of monotonicity in order to facilitate the analysis and design of efficient algorithms for checking m-privacy.

EG monotonicity is more restrictive than generalization monotonicity. If a constraint is EG monotonic, it is also generalization monotonic. But vice versa does not always hold. K-Anonymity and l-diversity that requires l distinct values of sensitive attribute in an equivalence group are examples of EG monotonic constraints, which are also generalization monotonic. Entropy l-diversity and t-closeness are examples of generalization monotonic constraints that are not EG monotonic at the same time. For example, consider a subset of two anonymized records with 2 different sensitive values satisfying entropy l-diversity (l = 2), i.e. distribution of sensitive attribute values in the group is uniform. Entropy l-diversity is not EG monotonic because it will not hold if we add a record that will change the distribution of sensitive values (and

entropy) significantly. However, it is generalization monotonic because it will still hold if any other subgroup satisfying entropy l-diversity (l = 2) is added (generalized) into the first subgroup.

Observation 2.2: If all constraints in a conjunction C = C1 ∧ C2 ∧ . . . ∧ Cw are EG monotonic, then the constraint C is EG monotonic. Similar observation holds for generalization monotonicity. In our example, C is defined as a conjunction of k-anonymity and l-diversity. Since both of them are EG monotonic [9], C is EG monotonic.

Theorem 2.1: m-Privacy with respect to a constraint C is EG monotonic if and only if C i s EG monotonic. Due to limited space, the proof of this theorem as been moved.

Observation 2.3: If a constraint C is EG monotonic, then A definition of m-privacy w.r.t. C(Definition 2.1) may be simplified. T * = A (T) satisfies m-privacy w.r.t. C, if and only if,

∀I ⊂ P, |I | = m, C is monotonic, C (A (T \TI)) = true

Indeed, for an EG monotonic C , if a coalition I cannot breach privacy, then any sub-coalition with fewer records cannot do so either (Definition 2.3). Unfortunately, generalization monotonicity of C is not sufficient for the simplification presented in this observation.

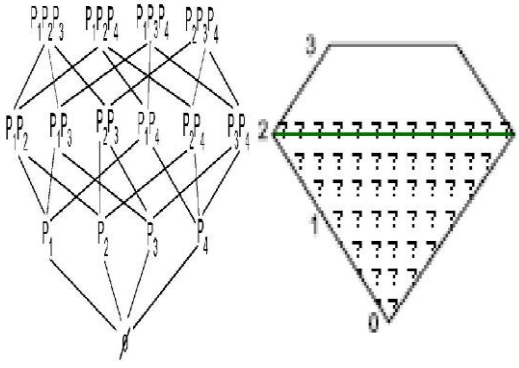


Figure 3: M-Adversary space.

Pruning Strategies

The pruning strategies are possible thanks to the EG monotonicity of m-privacy (Observations 2.1, 2.3). If a coalition is not able to breach privacy, then all its sub-coalitions will not be able to do so and hence do not need to be checked (downward pruning). On the other hand, if a coalition is able to breach privacy, then all its super-coalitions will be able to do so and hence do not need to be checked (upward pruning). In fact, if a sub-coalition of an m-adversary is able to breach privacy, then the upward pruning allows the algorithm to terminate immediately as the m-adversary will be able to breach privacy (early stop). Figure 3 illustrates the two pruning strategies where + represents a case when a coalition does not breach privacy and − otherwise.

The Top-Down Algorithm

The top-down algorithm checks the coalitions in a top-down fashion using downward pruning, starting from (nG − 1)-adversaries and moving down until a violation by an m-adversary is detected or all m-adversaries are pruned or checked.

The Bottom-Up Algorithm

The bottom-up algorithm checks coalitions in a bottom up fashion using upward pruning, starting from 0-adversary and moving up until a violation by any adversary is detected (early-stop) or all m-adversaries are checked.

The Binary Algorithm

The binary algorithm, inspired by the binary search algorithm, checks coalitions between (nG − 1) - adversaries and m-adversaries and takes advantage of both upward and downward pruning (Figure 5, Algorithm 1). The goal of each iteration is to search for a pair Isub and Isuper, such that Isub is a direct sub-coalition of Isuper and Isuper breaches privacy while Isub does not. Then Isub and all its sub-coalitions are pruned (downward pruning),

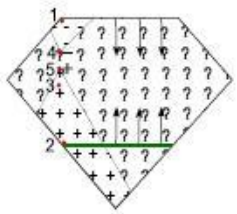


Fig. 5. The *binary* verification algorithm.

Adaptive Selection of Algorithms

Each of the above algorithms focuses on different search strategy, and hence utilizes different pruning. Which algorithm to use is largely dependent on the

characteristics of a given group of providers. Intuitively, the privacy fitness score (Equation 1), which quantifies the level of privacy fulfilment of records, may be used to select the most suitable verification algorithm. The higher the fitness score of attacked records, the more likely m-privacy will be satisfied, and hence a top-down algorithm with downward pruning will significantly reduce the number of adversary checks. We utilize such an adaptive strategy in the anonymization algorithm (discussed in the next section) and will experimentally compare and evaluate different algorithms in the experiment section.

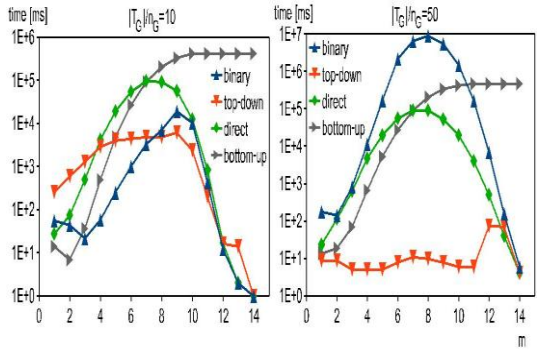
Adaptive m-privacy verification

M-Privacy is then verified for all possible splitting points and only those satisfying M - privacy are added to a candidate set π' (line 4). In order to minimize the time, our algorithm adaptively selects an m- privacy verification strategy using the fitness score of the partitions. Intuitively, in the early stage of the anonymization algorithm, the partitions are large and likely m-private. A top-down algorithm, which takes advantage of the downward pruning, may be used for fast verification. However, as the algorithm continues, the partitions become smaller, the down-ward pruning is less likely and the top-down algorithm will be less efficient. A binary algorithm or others may be used instead to allow upward pruning. We experimentally determine the threshold of privacy fitness score for selecting the best verification algorithm and verify the benefit of this strategy. Privacy Fitness Score Based Splitting Point Selection. Given a non-empty candidate set privacy fitness score (Definition 3.1) defined in the previous section and chooses the best splitting point (line 8). Intuitively, if the resulting partitions have higher fitness scores, they are more likely to satisfy m-privacy with respect to the privacy constraint and allow for further splitting. We note that the fitness score does not have to be exactly the same function used for adaptive ordering in m-privacy check. For example, if we use Equation 1, the weight parameter used to balance fitness values of privacy constraints, should have, most likely, different value. The algorithm then splits the partition and runs recursively on each sub-partition (line 9 and 10).

Name	Description	Verification	Anonymization
α	Weight paramter	0.3	0.8
m	Power of m-privacy	5	3
n	Total number of data providers	–	10
nG	Number of data providers contributing to a group	15	–
|T |	Total number of records	–	45,222
|TG |	Number of records in a group	{150, 750}	–
k	Parameter of k-anonymity	50	30
l	Parameter of l-diversity	4	4

B. m-Privacy Verification

The objective of the first set of experiments is to evaluate the efficiency of different algorithms for m-privacy verification given a set of records TG with respect to the previously defined privacy constraint C. Attack Power. In this experiment, we compared the different m-privacy verification heuristics against different attack powers. We used two different groups of records with relatively small and large average number of records per data provider, respectively. Figure 6 shows the runtime with varying m for different heuristics for the two groups.



Number of Contributing Data Providers. In this experiment, we analyzed the impact of contributing data providers (nG) on the different algorithms for the small and large group respectively. Figure 7 shows the runtime of different heuristics with varying number of contributing data providers' nG.

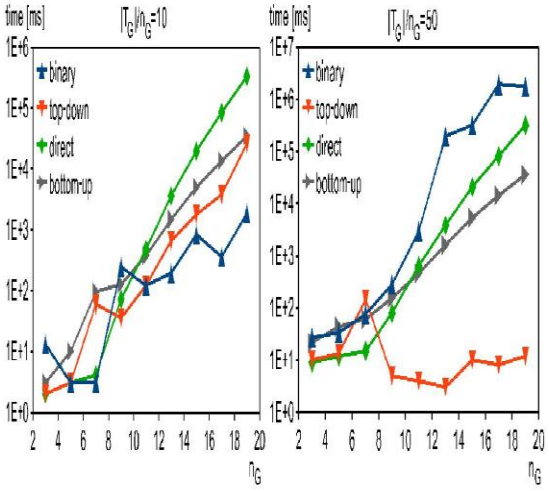


Figure 6: Runtime (logarithmic scale) vs. number of data providers.

This set of experiments compares our provider-aware algorithm with the baseline algorithm and evaluates the benefit of provider-aware partitioning as well as the adaptive m- privacy verification on utility of the data as well as efficiency. To evaluate the utility of the anonymized data, we used the query error metric similar to prior work (e.g. [18], [19]). 2,500 queries have been randomly generated and each query had qd predicates pi , defining a range of a randomly chosen quasi-identifier, where qd $\in [2, q]$ and q is the number of quasi-identifier attributes.

SELECT t FROM T* WHERE p1 AND . . AND pqd ;
Query error is defined as the difference in the results coming from anonymized and original data. Attack Power. We first evaluated and compared the two algorithms with varying attack power m. Figure 9 shows the runtime with varying m for the two algorithms respectively. We observe that the provider-aware algorithm significantly outperforms the baseline algorithm. This fact may look counter intuitive at the first glance – our algorithm considers one more candidate splitting point at each iteration, thus the execution time should be higher. However, in each iteration of the provider-aware algorithm, the additional splitting point along data providers, if chosen, reduces the number of providers represented in a subgroup and hence reduces m-privacy verification time significantly (as observed in Figure 7). In contrast, the baseline algorithm preserves the average number of providers in each subgroup, which incurs a high cost for m-privacy verification. As expected, both algorithms show a peak cost when m ≈ n/2.

4. CONCLUSION

In this paper, we considered a new type of potential attackers in collaborative data publishing a coalition of data providers, called m-adversary. To prevent privacy disclosure by any m-adversary we showed that guaranteeing m-privacy is enough. We presented heuristic algorithms exploiting equivalence group mono tonicity of privacy constraints and adaptive ordering techniques for efficiently checking m-privacy. We introduced also a provider-aware anonymization algorithm with adaptive m-privacy checking strategies to ensure high utility and m-privacy of anonymized data. Our experiments confirmed that our approach achieves better or comparable utility than existing algorithms while ensuring m-privacy efficiently.

There are many remaining research questions. Defining as per privacy fitness score for different privacy constraints is one of them. It also remains a question to address and model the data knowledge of data providers when data are distributed in a vertical or ad-hoc fashion. It would be also interesting to verify if our methods can be adapted to other kinds of data such as set-valued data.

5. ACKNOWLEDGMENT

This work is supported by a Cisco Research Award. The authors would also like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

REFERENCES

[1] C. Dwork, "Differential privacy: a survey of results," in Proc. of the 5th Intl. Conf. on Theory and Applications of Models of Computation, 2008, pp. 1–19.
[2] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," ACM Computer. Surv, vol. 42, pp. 14:1–14:53, June 2010.
[3] C. Dwork, "A firm foundation for private data analysis," Commun. ACM, vol. 54, pp. 86–95, January 2011.

[4] N. Mohammed, B. C. M. Fung, P. C. K. Hung, and C. Lee, "Centralized and distributed anonymization for high-dimensional healthcare data," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 4, no. 4, pp. 18:1–18:33, October 2010.

[5] W. Jiang and C. Clifton, "Privacy-preserving distributed k-anonymity," in Data and Applications Security XIX, ser. Lecture Notes in Computer Science, 2005, vol. 3654, pp.924–924.

[6] W. Jiang and C. Clifton, "A secure distributed framework for achieving k-anonymity," VLDB J., vol. 15, no. 4, pp. 316–333, 2006.

[7] O. Gold Reich, Foundations of Cryptography: Volume 2, Basic Applications. Cambridge University Press, 2004.

[8] Y. Lindell and B. Pinkas, "Secure multiparty computation for privacy preserving data mining," The Journal of Privacy and Confidentiality, vol. 1, no. 1, pp. 59–98, 2009.

[9] Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkita subramaniam, "l-diversity: Privacy beyond k-anonymity," in ICDE, 2006, p. 24.

[10] P. Samarati, "Protecting respondents' identities in micro data release," IEEE T. Knowl. Data En, vol. 13, no. 6, pp. 1010–1027, 2001.

[11] L. Sweeney, "k-anonymity: a model for protecting privacy," Int. J.Uncertain. Fuzz, vol. 10, no. 5, pp. 557–570, 2002.

[12] N. Li and T. Li, "t-closeness: Privacy beyond k-anonymity and ldiversity," in In Proc. of IEEE 23rd Intl. Conf. on Data Engineering (ICDE), 2007.

[13] R. Burke, B. Mobasher, R. Zabicki, and R. Bhaumik, "Identifying attack models for secure recommendation," in In beyond Personalization: a Workshop on the Next Generation of Recommender Systems, 2005.

[14] D. Kifer, "Attacks on privacy and define tti's theorem," in Proc. of the 35th SIGMOD Intl. Conf. on Management of Data, 2009, pp. 127–138.

[15] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," in Proc. of the 2011 Intl. Conf. on Management of Data, 2011, pp. 193–204.