International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I10P204

Volume 70 Issue 10, 27-34, October 2022

Short Communication

Towards Privacy Preserving Data Publishing in Inter Cloud Infrastructure

Veena Gadad[1], C. N. Sowmyarani[2]

[1, 2]*Department of Computer Science and Engineering, R V College of Engineering, Bangalore, Karnataka, India.*

[1]*Corresponding Author : veenagadad@rvce.edu.in*

Received: 17 July 2022 Revised: 25 September 2022 Accepted: 05 October 2022 Published: 19 October 2022

Abstract *- Data privacy is a prime concern in this digital era since an enormous amount of data is collected, stored and published regularly. Due to gratifying features like data sharing, easy maintenance, economical, large network access and fast processing, many organizations and users leverage the cloud environment for data storage and access. However, when such an environment is used for data publishing, there are chances of an individual's identity and sensitive information leakage. These are caused by the external attacker and the internal cloud environment. Privacy Preserving Data Publishing (PPDP) is a suite of anonymization algorithms that aim to prevent such attacks while simultaneously safeguarding the person's identity. Studies have shown that popular privacy algorithms like p sensitive k-anonymity, KP cover and differential privacy, though they provide stronger privacy, are less efficient in preventing emerging attacks. This paper proposes a novel algorithm to publish data in the public cloud and prove that it is computationally efficient and prevents privacy attacks that are especially caused by the data published in the cloud environment.*

Keywords *- Data Privacy, Privacy attacks, anonymization, PPDP, Differential Privacy, Cloud data privacy.*

1. Introduction

Data collection is the first process in any organization. The collected data is accumulated, processed and published to the outside world. This is shown in Fig. 1.

The data gets collected at hospitals, schools, companies, and e-commerce and web portals. Commonly collected data include name, age, zip code, Gender, medical data, travel details, nature of the job, address etc. When observed, most of these data contain sensitive information that is individual-specific.

The organizations that collect data adopt an intercloud environment for data storage. The reasons are:(i) Data is generated faster as the internet-associated gadgets increase. (ii)With raising data usage, the user's requirement cannot be satisfied with the local machine's capacity. (iii) The cloud offers features like storage, fast computing, and economical and easy maintenance[1].

As per the mentioned reasons, it is clear that for any organization, there is a guaranteed need to store and manage the data in inter-cloud infrastructure while preserving sensitive information from privacy breaches.

The data collected at any organization is the driving force to make major decisions that must be filtered, sorted, processed and analysed. Therefore, it has to be published in the cloud so that the data can be utilized by researchers, data scientists, and big data analysts to test and try various techniques on the data.

Apart from constructive usage of the published data, there might be an intruder who causes harm to individuals by gaining their sensitive information from the published data [5]. As per the study, 120 nations worldwide have some form of international privacy for data protection [2] to ensure data protection and controls. But, just enforcing the laws is insufficient to provide data privacy; there is a high need for algorithms and frameworks to support the laws.

The data processed and published is in the microdata table (data specific to an organization). This microdata consists of *'r'* records with values specific to *'a'* attributes. Table 1 shows the sample data that is collected at a healthcare organization. The table consists of various attributes such as Identifiers (ID), QuasiIdentifiers (QID's), Sensitive Attributes (SA's) and Non-Sensitive Attributes (NSA's).

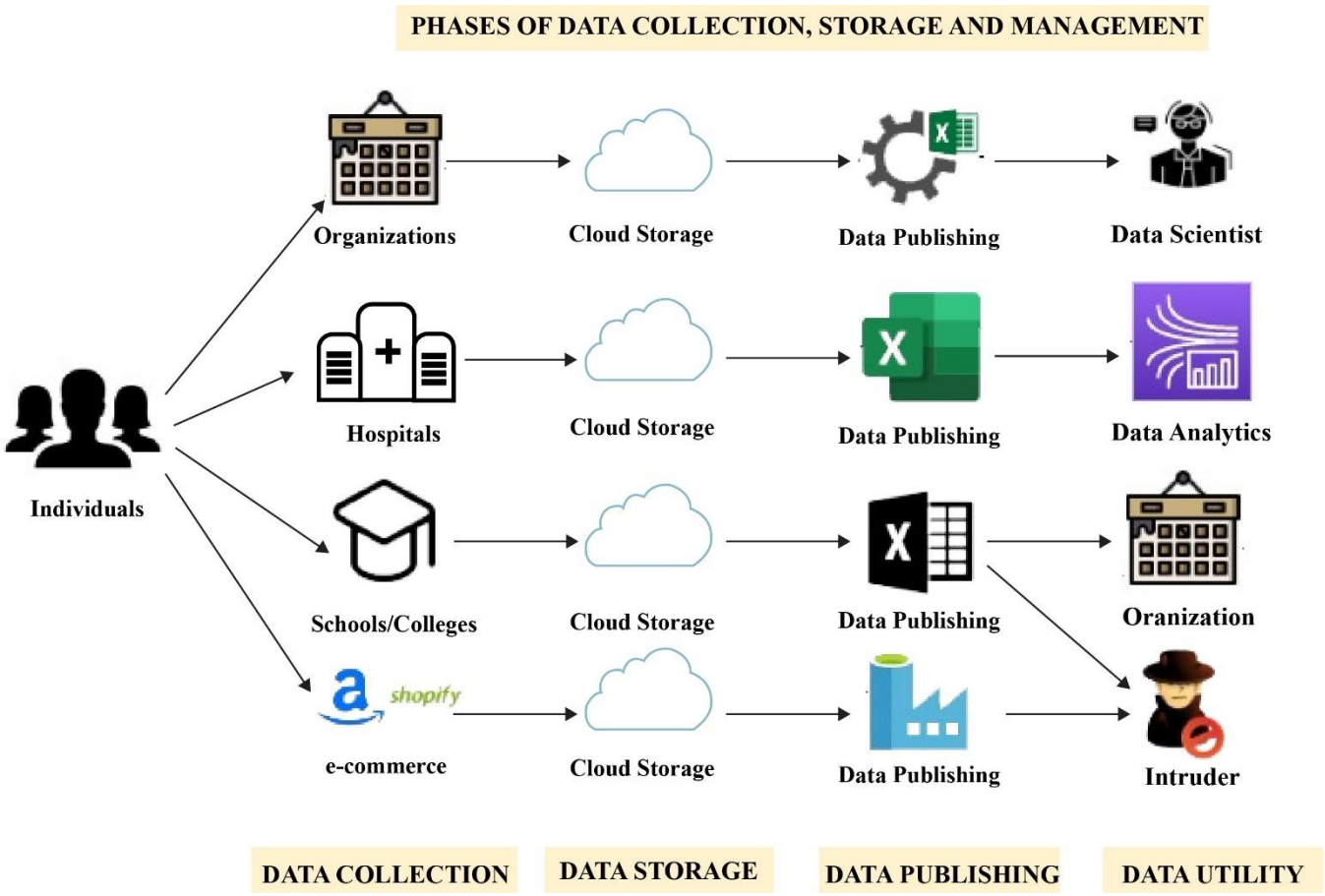

Fig. 1 Different phases of data collection

Table 1. Sample Health Care data

ID	QID's			SA's		NSA's
Patient Name	***Pin Code***	***YOB***	***Gender***	***Medical Condition***	***Salary***	***Extra Activities***
Rohini	570889	21-09-2001	F	Cancer	12	Travelling
Rahul	570899	23-06-1988	M	Heart Disease	24	Playing
Rashmi	570819	24-08-1999	F	Cancer	32	Music
Maddy	570802	11-09-1987	M	Uremia	12	Vocal
Kemu	570819	13-05-1998	M	Dementia	20	Trekking

According to the literature [2], the attributes are defined as follows:

1. *ID*- These attributes identify the individual directly. It could be *Name, Patient ID and Unique Number*. Organizations remove these identifying attributes before sharing the data.
2. *QID's*- Some attributes, combined with attributes of other published data, may reveal identity. Such attributes are called quasi-attributes or QIDs. Namely, *age, nature of work,* zip *code and Gender.*
3. *SA's*- Personal information such as *medical condition/treatment, travel history, salary, marital status and relationship*, which are individually specific, are called sensitive attributes.
4. *NSA's*- Some attributes may not be sensitive. However, NSA may contribute to re-identify the individuals. These are termed non-sensitive attributes.

Data Privacy refers to an individual's right to know what information is stored about them, control how the information is communicated and prevent its unwanted use. Since the public cloud service providers (CSPs) are untrusted and curious [3], achieving data privacy is more challenging because of the following observed reasons:

1. It is impossible to determine the physical location of the cloud servers or configure the information processing.
2. Cloud data can be modified and replicated easily by the CSP.

3. Development of new business/service models and their implications for consumer privacy.
4. Losing control over personal data is much easier, leading to major privacy threats.

The CSPs have access to a large amount of sensitive data and perform various analyses. In some scenarios, data security and privacy are achieved through encryption-based methods [4] [5]. These methods have the following drawbacks: 1. The algorithms are computationally costlier and less efficient against internal attacks. 2. Since these algorithms require a key for encryption or decryption, the utility is less.

A plethora of anonymization algorithms has been proposed in the literature to preserve data privacy. K-Anonymity [1] was the first and simple anonymization technique for preserving the privacy of single sensitive attributes; here, k indicates the number of records grouped into one class where QID values are the same in all k records.

The paper discusses a novel algorithm based on the concept of Anatomy [7]. The original table is vertically partitioned into separate QID tables and SA table/s. The SA table/s are further partitioned horizontally and given the group id according to k-anonymity criteria. Finally, the anonymized table is published in the cloud storage. The advantages of the proposed algorithm are:

1. The computation cost is less as compared to popular anonymization algorithms.
2. The algorithm provides data privacy against emerging attacks in the cloud environment.

1.1. Organization of the paper

Section II discusses the related work. Section III presents the methodology to achieve data privacy in the cloud. Section IV provides results and discussion. Finally, the paper concludes with Section V.

2. Related Work

Data protection can be achieved by encryption or anonymization techniques. However, there is a difference between data encryption and data anonymization [4]. Data encryption includes encryption and decryption keys. Anyone who wants to encrypt or decrypt the data must use the keys. The main objectives of encryption algorithms are Confidentiality, Integrity and Access control. Anonymization is treating the data to prevent disclosures. Here, various methods like a generalization, suppression, anatomization, slicing, permutation, aggregation, and noise addition are applied to the data. The advantage of anonymization techniques for data protection is that anyone can use the data without requiring the key. The main objectives of data anonymization are to provide data protection and maximize data utilization. Any statistical results obtained after applying the anonymized data must almost be the same as those obtained on the original data.
The existing algorithms in the literature can be classified and studied based on the operations used for data anonymization.

2.1. Generalization and Suppression

These are the common operations used [8] [9]. The generalization operation replaces the QID values with broader domain values. For example: For the categorical attribute Occupation, if the original value is Cardiologist, it is generalized to Doctor. Similarly, for the numerical attribute Age, if the value is 30, it is generalized to a range of values as 20-40. Generalization is also known as recoding, and there are two types of recoding: Local and global, depending upon the nature of generalizations. Suppression suppresses the attribute values with symbolic characters like *and # and anonymizes the data. Many of the anonymization algorithms' generalizations and suppressions are combined to anonymize the data. There are two main drawbacks with the algorithms that use generalizations and suppressions: i. These algorithms consider single sensitive attributes (such as disease/salary), and extending them to provide multiple sensitive attribute protection is difficult. ii. Information loss may lead to lesser utility in generalising or suppressing.

2.2. Anatomization and Slicing

In these operations, original values of the attributes are retained, and separate tables of SA and QIDs are generated. The main advantage of these operations is that they are less information loss and perform better with high dimensional data [10]. The anatomy [7] algorithm releases the QID and SA tables separately. These tables have a common identifier called Gropu_ID. The algorithm achieves privacy since it is impossible to identify any tuple's sensitive attribute. Slicing [11] divides the microdata vertically as anatomy also horizontally. The algorithm satisfies the privacy requirement of l-diversity and tries to preserve the correlation between the SA and QID. The horizontal partition results in the creation of buckets. The buckets contain a subset of records. The vertical partition results in separate tables where each table contains a subset of attributes. The algorithm also releases multiple tables with the original values. In [12], the t-closeness slicing algorithm was designed to preserve privacy in transactional data and overcome privacy attacks. The anonymization algorithm in [26] satisfies both k-anonymity and l-diversity. It also uses anatomy and slicing to preserve the privacy of multiple sensitive attributes.

2.3. Permutation

Permutation follows the same approach as anatomization, i.e., it disassociates the relationship between the QID and SA, however it further groups the tuples into groups and then re-arranges the sensitive values within each group [14][15]. Permutation anonymization [16] divides the microdata table so that each group satisfies l-diversity. A lightweight privacy algorithm is proposed in [17] [18]. It uses a pseudo-random permutation to alter the order of the data.

2.4. Perturbation

These operations are applied to the dataset to distort the data to protect privacy by preserving the statistical properties. Perturbation is achieved by swapping or adding noise to the data. Microaggregation or generating synthetic data is also a type of perturbation technique. Differential privacy [27] incorporates random noises into the data so that an intruder receives imprecise data, and it becomes difficult to breach privacy. In Data swapping, sensitive values are exchanged to protect the statistical information [20].
Microaggregation [21] basically groups the records satisfying the k-anonymity or l-diversity [22] and then replace the records in a group by aggregate values such as mean, median or standard deviation.

3. Methodology

Though the cloud offers many beneficial features, as mentioned earlier, data owners must be careful when the data is published in such environments. Figure 2 presents the flowchart for privacy-preserving data publishing in the cloud.

The data owner identifies the Quasi Identifiers (QIDs) and Sensitive Attributes (SAs) from the collected data table that needs to be stored in the cloud. Examples of such data include Health care data, Educational Institutes data and E-commerce data.

The data is vertically partitioned into separate QID tables (QT) and SA tables (SAT). The purpose of such a partition is to break the correlation between the attributes so as to prevent disclosures.

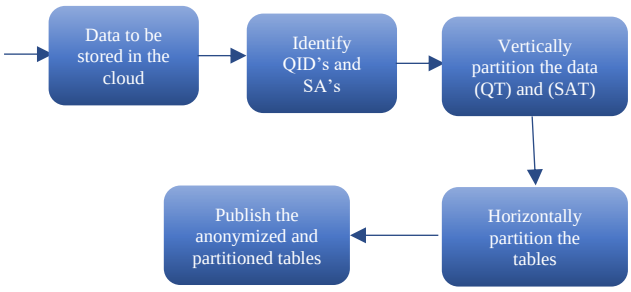

Fig. 2 Flowchart for Privacy Preserving Data Publishing in the cloud

Table 2. Notations used in the algorithm

Notation	Meaning
NS1, NS2,..NSn	Non-Sensitive Attributes.
Sa1,Sa2,..San	Sensitive Attributes
ST	Sensitive Attribute Table
ST1, ST2,..STn	Partitioned Sensitive Attribute Tables
Qd1,Qd2,..Qdn	Quasi Identifiers
QT	Quasi Identifier Table
K	Partition criteria

The Mondrian partitioning algorithm [24] is applied to the SA table, and the partitions are created. Each partition is assigned with the group id. The intention of creating this partition is that when the anonymized table is constructed, the tuples in different tables are identified through these group ids. Table 2 provides the notations used in the algorithms. Figure 3. Provides the proposed algorithm.

Once the AN tables and SA tables are published separately, the system accepts the end user aggregate query, providing the results.

To understand the proposed methodology, consider the sample microdata table after removing the identifying attributes, shown in Table 4.

Algorithm: Partition

Input: D*ata table T.*

Output: *Vertically and Horizontally Partitioned QT and ST tables.*

1. Input Table T*(Qd1,Qd2,..Qdn, Sa1,Sa2,..San, NS1, Ns2..Nsn).*
2. Identify the attributes in the table as quasi identifiers *(Qd1,Qd2,..Qdn)* and sensitive attributes*(Sa1,Sa2,..San).*
3. Construct the separate tables QT and ST by extracting the QIDs and SA, respectively. *QT(Qd1,Qd2,..Qdn), ST(Sa1,Sa2,..San).*
4. For every Sa's in ST, determine the dependency with other Sa's according to the dependency table shown in Table 3.
    a. The dependent Sa's of ST are separated and extracted into new tables *ST1 and ST2. STn.*
    b. Choose the horizontal partition criteria,k.
    c. For each ST, horizontally partition the table according to Mondrian Partitioning Algorithm and chosen k.
    d. For each partition created, assign the group id's-0,1,2….N.
5. Construct the anonymized table AT(*Qd1, Qd2,..Qdn, G1, G2..Gn)* by combining the QIDs and the group-id's of each tuple referring to the partitioned ST's.
6. Create cloud storage containers (buckets/blobs) to store the tables.
7. Publish the AN and ST tables in the cloud storage.

Fig. 3 Proposed algorithm 'Partition.'

Table 3. The Dependency table

Sensitive attributes	Dependency
Marital Status	-
Relationship	Marital Status
Education	Salary
Salary	Relationship

The QIDs and SA are identified. QID's: Age, Gender and Zip code. SA: Marital Status, Relationship and Salary. For this table, the dependency table Table 3 is constructed.

Tables 5 and 6 show the partitions created using Algorithm 2 with partition criteria as k-anonymity. The final anonymized table shown in Table 7 is constructed using the group IDs in Tables 5 and 6.

Table 4. Sample data table

Tuple ID	Age	Gender	Zip Code	MaritalStatus	PersonalRelationship	Salary	Education
0	39	Male	77516	Never-married	Not-in-family	<=50K	Bachelors
1	50	Male	83311	Married-civ-spouse	Husband	<=50K	Bachelors
2	38	Male	215646	Divorced	Not-in-family	<=50K	HS-grad
3	53	Male	234721	Married-civ-spouse	Husband	<=50K	Bachelors
4	28	Female	338409	Married-civ-spouse	Wife	<=50K	11^{th}
5	37	Female	284582	Married-civ-spouse	Wife	<=50K	Masters
6	49	Female	160187	Married-spouse-absent	Not-in-family	<=50K	9^{th}
7	52	Male	209642	Married-civ-spouse	Husband	>50K	HS-grad
8	31	Female	45781	Never-married	Not-in-family	>50K	Masters
9	42	Male	159449	Married-civ-spouse	Husband	>50K	Bachelors

Table 5. Partitioned Sensitive Attribute table (SA1)

Education	Salary	G1
HS-grad	>50K	0
Masters	>50K	0
Bachelors	>50K	0
Bachelors	<=50K	1
Bachelors	<=50K	1
HS-grad	<=50K	1
Bachelors	<=50K	1
11^{th}	<=50K	2
Masters	<=50K	2
9^{th}	<=50K	2

Table 6. Partitioned Sensitive Attribute table (SA2)

MaritalStatus	Personal Relationship	G2
Never married	Not-in family	0
Divorced	Not-in family	0
Married-spouse-absent	Not-in family	0
Never married	Not-in family	0
Married civ-spouse	husband	1
Married	husband	1
civ-spouse	Wife	1
Married	Wife	1
civ-spouse	husband	1
Married	husband	1

Table 7. Anonymized table

Tuple Id	Age	ZipCode	Gender	G1	G2
0	39	77516	Male	0	0
1	50	83311	Male	1	1
2	38	215646	Male	0	1
3	53	234721	Male	1	2
4	28	338409	Female	1	1
5	37	284582	Female	1	2
6	49	160187	Female	0	2
7	52	209642	Male	1	0
8	31	45781	Female	0	0
9	42	159449	Male	1	0

The anonymized table, and sensitive attribute tables are published in the cloud storage. The CSP and the published data user have access to the anonymized table and the partitioned sensitive attribute tables, which protects data privacy. In the next section, the performance of the proposed algorithm and its efficacy against privacy attacks is discussed.

4. Results and Discussion

To test the proposal, we considered the real data set containing the demographic data, Adult data set from the UCI machine learning repository [25] with 30162 records. The *Partition* algorithm is implemented using Python 3.9.

Intel Core i3 Processor with 8 GB DDR4 RAM and 64, bit Windows 10 Operating System, and 256 GB of storage. Azure cloud storage is used for publishing the data. For experimentation, we have chosen the processing time to create the partitions and the number of partitions created as the parameters.

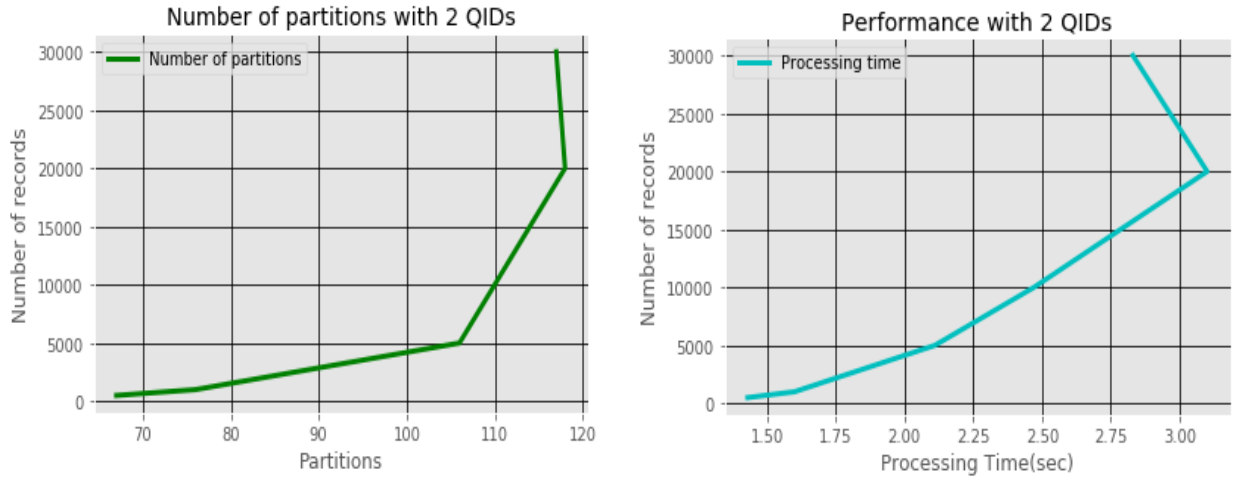

Fig. 5 Number of partitions and processing time with 2QIDs

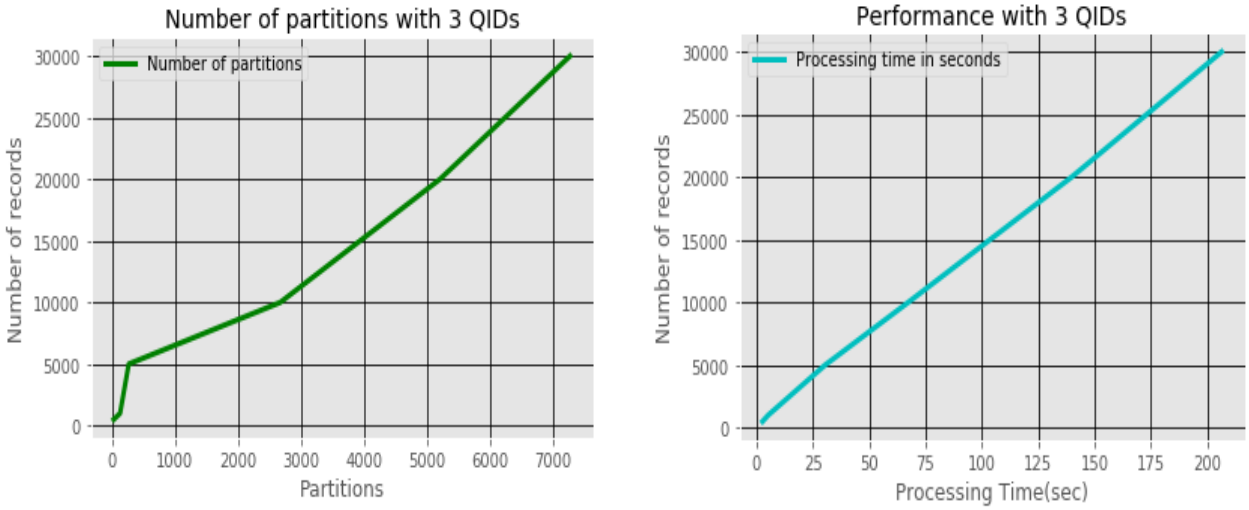

Fig. 6 Number of partitions and processing time with 3QIDs

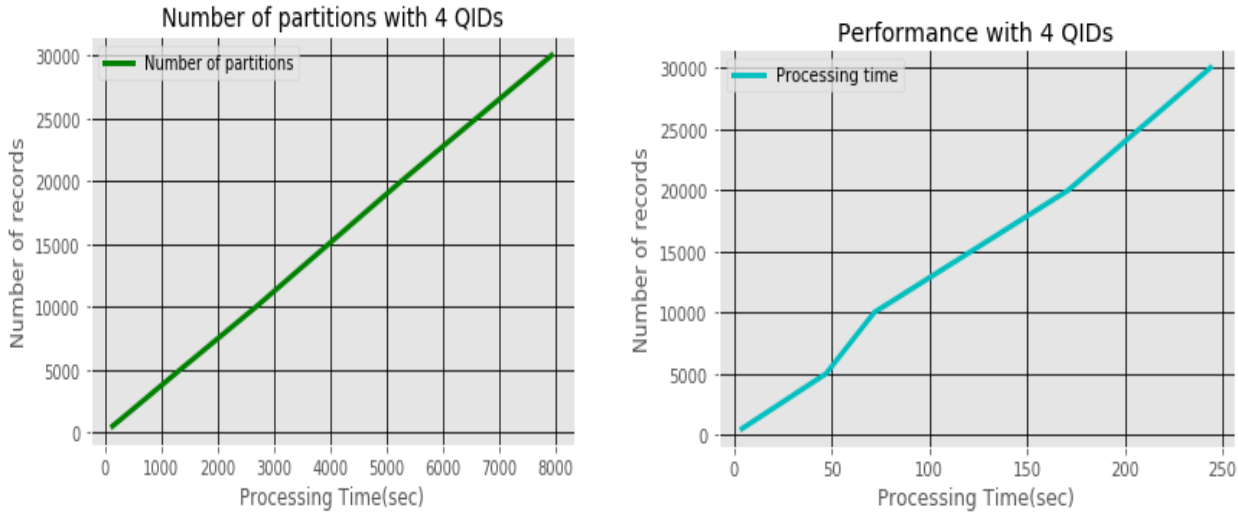

Fig. 7 Number of partitions and processing time with 4QIDs.

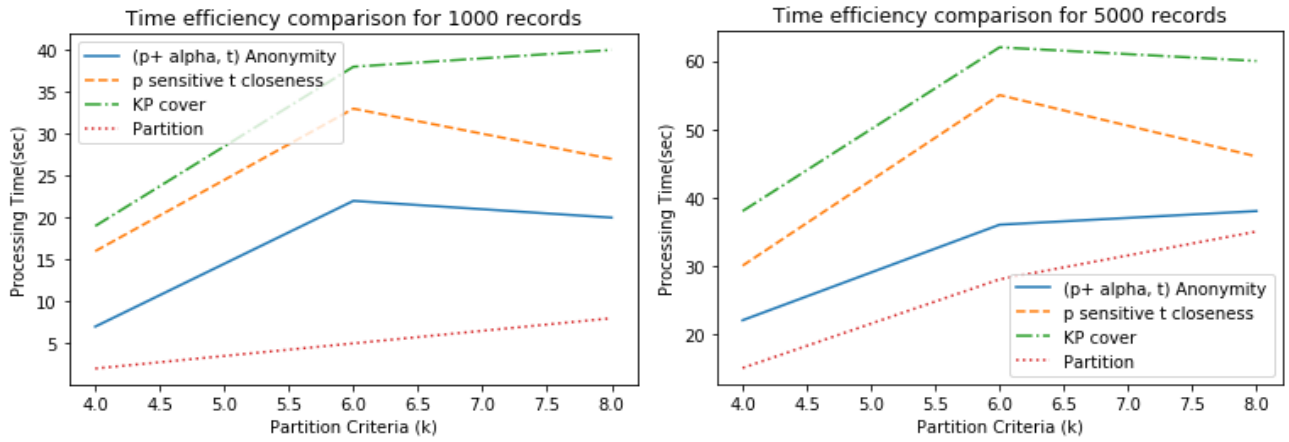

Fig. 8 Performance comparison of various algorithms

4.1. Performance Measurement

The dataset has 14 attributes. Fig. 5 shows the number of partitions created and the processing time when 2 QIDs were chosen (Age, Zip Code). Fig. 6 showed the variations of parameters when 3 QIDs were chosen (Age, Zip code and Gender). Fig. 7 shows the parameter variations when considering 4 QIDs (Age, ZipCode, Gender and Employment).

For all the cases, the partition criteria, k, is chosen to be 3. From the results, it can be inferred that the proposed algorithm works well with any number of sensitive attributes.

The number of partitions is created appropriately per the partition criteria when the QIDs are more than 2.

As mentioned in the related work section, many anonymization algorithms have been implemented and tested. The proposed algorithm is compared with three popular anonymization algorithms :(p+ alpha, t) anonymity, p-sensitive t-closeness and KP cover. These results are discussed in detail in [23].

Fig. 8 shows the time efficiency of the Partition algorithm in comparison with the other algorithms. As seen, the processing time is comparatively less when compared to existing techniques.

4.2. Privacy Attacks

The proposed algorithm is efficient against background knowledge attacks, multiple sensitive correlation attacks and identity disclosure attacks.

Consider the following scenario for understanding these attacks: From table 7, the user can infer that a person with age 50, zip code 83311, has done his Bachelor/HS-grad with Salary<=50K and is Married. It shows that an individual's sensitive attributes are not revealed directly and that the technique overcomes the background knowledge attack.

With the *Partition* algorithm, the original table is partitioned vertically as well as horizontally. This eliminates the correlation between the quasi-identifiers and sensitive attributes. The algorithm, therefore, overcomes any type of correlation attack. Also, since the algorithm disassociates the QIDs and the SAs, if the attacker knows any of the QIDs, he will not be able to identify the exact record of the person

5. Conclusion

The paper discusses a novel privacy-preserving data publishing algorithm in inter-cloud architecture to prevent privacy breaches that may take place by internal or external attackers. The algorithm is computationally efficient when compared with well-known anonymization techniques. In the experiments conducted, we proved that the algorithm is time efficient and prevents privacy threats in the cloud environment. As a part of future work, we plan to analyse the algorithm's performance with aggregate query analysis on the published data.

References

[1] P. Mell, T. Grance, and Others, 'the Nist Definition of Cloud Computing', 2011.

[2] "Beyond Gdpr: Data Protection Around the World," *Thales Group*, 10-May-2021.[Online].Available: https://www.Thalesgroup.Com/EN/Markets/Digital-Identity-and-Security/Government/Magazine/Beyond-Gdpr-Data-Protection-Around-World. [Accessed: 06-May-2022].

[3] J. Domingo-Ferrer, O. Farràs, J. Ribes-González, and D. Sánchez, "Privacy-Preserving Cloud Computing on Sensitive Data: A Survey of Methods, Products and Challenges," *Computer Communications*, . Elsevier Bv, vol. 140–141Pp. 38–60, May 2019. Doi: 10.1016/J.Comcom.2019.04.011.

[4] P. K. P, S. K. P, and A. P.J.A., "Attribute Based Encryption in Cloud Computing: A Survey, Gap Analysis, and Future Directions," *Journal of Network and Computer Applications*, vol. 108. Elsevier Bv, pp. 37–52, 2018. Doi: 10.1016/J.Jnca.2018.02.009.

[5] N. Kaaniche and M. Laurent, "Data Security and Privacy Preservation in Cloud Storage Environments Based on Cryptographic Mechanisms," *Computer Communications,"* Elsevier Bv, vol. 111, pp. 120–141, 2017. Doi: 10.1016/J.Comcom.2017.07.006.

[6] Sowmyarani C. N. and Dayananda P, "Analytical Study on Privacy Attack Models in Privacy Preserving Data Publishing," *Security Solutions and Applied Cryptography in Smart Grid Communications.* IGI Global, pp. 98–116. Doi: 10.4018/978-1-5225-1829-7.Ch006

[7] X. Xiao Και Y. Tao, "Anatomy: Simple and Effective Privacy Preservation," *Proceedings of the 32nd International Conference on Very Large Data Bases*, pp. 139–150, 2006.

[8] P. Samarati K, L. Sweeney, "Protecting Privacy When Disclosing Information: K-Anonymity and Its Enforcement Through Generalization and Suppression," 1998.

[9] K. Lefevre, D. J. Dewitt, Και R. Ramakrishnan, "Incognito: Efficient Full-Domain K-Anonymity," *Proceedings of the 2005 ACM Sigmod International Conference on Management of Data*, pp. 49–60, 2005.

[10] C. C. Aggarwal, "On K-Anonymity and the Curse of Dimensionality," ΣTO VLDB, 2005, vol. 5, pp. 901–909.

[11] T. Li, N. Li, J. Zhang, K, I. Molloy, "Slicing: A New Approach for Privacy-Preserving Data Publishing*," IEEE Transactions on Knowledge and Data Engineering*, vol. 24, pp. 561–574, 2010.

[12] M. Wang, Z. Jiang, Y. Zhang, Και H. Yang, "T-Closeness Slicing: A New Privacy-Preserving Approach for Transactional Data Publishing," *INFORMS Journal on Computing*, vol. 30, pp. 438–453, 2018.

[13] Andrea Li, "Privacy, Security and Trust Issues in Cloud Computing," *SSRG International Journal of Computer Science and Engineering*, vol. 6, no. 10, pp. 29-32, 2019. Crossref, https://doi.org/10.14445/23488387/IJCSE-V6I10P106.

[14] Y. Tao, H. Chen, X. Xiao, S. Zhou, Και D. Zhang, "Angel: Enhancing the Utility of Generalization for Privacy-Preserving Publication," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, 7, pp. 1073–1087, 2009.

[15] Q. Zhang, N. Koudas, D. Srivastava, T. Yu, "Aggregate Query Answering on Anonymized Tables," *2007 IEEE 23rd International Conference on Data Engineering*, pp. 116–125, 2007.

[16] D. Li, X. He, L. Cao, Και H. Chen, "Permutation Anonymization," *Journal of Intelligent Information Systems*, vol. 47, no. 3, pp. 427–445, 2016.

[17] M. Bahrami Και M. Singhal, "A Lightweight Permutation Based Method for Data Privacy in Mobile Cloud Computing," *2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering*, pp. 189–198, 2015.

[18] M. Bahrami, D. Li, M. Singhal, Και A. Kundu, "An Efficient Parallel Implementation of A Lightweight Data Privacy Method for Mobile Cloud Users," *2016 Seventh International Workshop on Data-Intensive Computing in the Clouds (Datacloud)*, pp. 51–58, 2016.

[19] Maryann Thomas, S. V. Athawale, "Study of Cloud Computing Security Methods: Cryptography," *SSRG International Journal of Computer Science and Engineering*, vol. 6, no. 4, pp. 1-5, 2019. *Crossref*, https://doi.org/10.14445/23488387/IJCSE-V6I4P101.

[20] S. E. Fienberg Και J. Mcintyre, "Data Swapping: Variations on A Theme By Dalenius and REISS," *Privacy in Statistical Databases*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 14–29, 2004.

[21] J. Domingo-Ferrer K, J. M. Mateo-Sanz, "Practical Data-Oriented Microaggregation for Statistical Disclosure, " *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, 1, pp. 189–201, 2002.

[22] H. Jian-Min, C. Ting-Ting, K, Y. Hui-Qun, "An Improved V-Mdav Algorithm for L-Diversity," *2008 International Symposiums on Information Processing*, pp. 733–739, 2008.

[23] C. N. Sowmyarani, V. Gadad, K, P. Dayananda, "(P+, A, T)-Anonymity Technique Against Privacy Attacks," *International Journal of Information Security and Privacy* (IJISP), vol. 15, no. 2, pp. 68–86, 2021.

[24] K. Lefevre, D. J. Dewitt, R. Ramakrishnan, "Mondrian Multidimensional K-Anonymity," *22nd International Conference on Data Engineering* (ICDE'06), pp. 25–25, 2006.

[25] C. Blake, "Uci Repository of Machine Learning Databases," http://www. ICS. Uci. Edu/\ Mlearn/Mlrepository. html, 1998.

[26] V. S. Susan,T. Christopher, "Anatomisation With Slicing: A New Privacy Preservation Approach for Multiple Sensitive Attributes," *Springerplus*, vol. 5, pp. 1–21, 2016.

[27] C. Dwork, "Differential Privacy: A Survey of Results," *International Conference on Theory and Applications of Models of Computation*, pp. 1–19, 2008.