Secure Anonymization Protocol: A Novel Protocol for Providing *m*-Privacy in Collaborative Data Publishing

Kalangi Vilson [#1], Dr.Koduganti Venkata Rao [*2]

M.Tech Scholar [#1], Professor & HOD *[2]

Department of Computer Science & Engineering,
Vignan Institute of Information Technology,
Visakhapatnam, AP, India.

.

Abstract

Data publishing/Data publication is a practice consisting in preparing certain data or data set(s) for public use thus to make them available to everyone to use as they wish. This practice is an integral part of the open science movement. There is a large and multidisciplinary consensus on the benefits resulting from this practice. The main goal is to elevate data to be first class research outputs. There are a number of initiatives underway as well as points of consensus and issues still in contention. In this paper, we consider the collaborative data publishing problem for anonymizing horizontally partitioned data at multiple data providers. We consider a new type of "insider attack" by colluding data providers who may use their own data records (a subset of the overall data) to infer the data records contributed by other data providers. This current paper mainly concentrated on the insider attack that collides the published data. This current issue is solved in 3 notions, First, we introduce the notion of *m*-privacy, which guarantees that the anonymized data satisfies a given privacy constraint against any group of up to *m* colluding data providers. Second, we present heuristic algorithms exploiting the monotonicity of privacy constraints for efficiently checking *m*-privacy given a group of records. Third, we present a data *provider-aware* anonymization algorithm with adaptive *m*-privacy checking strategies to ensure high utility and *m*-privacy of anonymized data with efficiency. By conducting several experiments on these three issues, we finally proposed a novel multiparty computation protocol for collaborative data publishing with m-privacy. Experiments on real-life datasets suggest that our approach achieves better or comparable utility and efficiency than existing and baseline algorithms while satisfying *m*-privacy.

Keywords

Clustering, Privacy Preserving, Data Security, Integrity, and Protection, Distributed Databases.

1. Introduction

Generally, data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to

analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases [1].

How Data Mining Works?

While large-scale information technology has been evolving separate transaction and analytical systems, data mining provides the link between the two. Data mining software analyzes relationships and patterns in stored transaction data based on open-ended user queries. Several types of analytical software are available: statistical, machine learning, and neural networks. Generally, any of four types of relationships are sought:

- **Classes**: Stored data is used to locate data in predetermined groups. For example, a restaurant chain could mine customer purchase data to determine when customers visit and what they typically order. This information could be used to increase traffic by having daily specials.
- **Clusters**: Data items are grouped according to logical relationships or consumer preferences. For example, data can be mined to identify market segments or consumer affinities.
- **Associations**: Data can be mined to identify associations. The beer-diaper example is an example of associative mining.
- **Sequential patterns**: Data is mined to anticipate behavior patterns and trends. For example, an outdoor equipment retailer could predict the likelihood of a backpack being purchased based on a consumer's purchase of sleeping bags and hiking shoes.

1.1 Data mining consists of five major elements:

1) Extract, transform, and load transaction data onto the data warehouse system.
2) Store and manage the data in a multidimensional database system.
3) Provide data access to business analysts and information technology professionals.
4) Analyze the data by application software.
5) Present the data in a useful format, such as a graph or table.

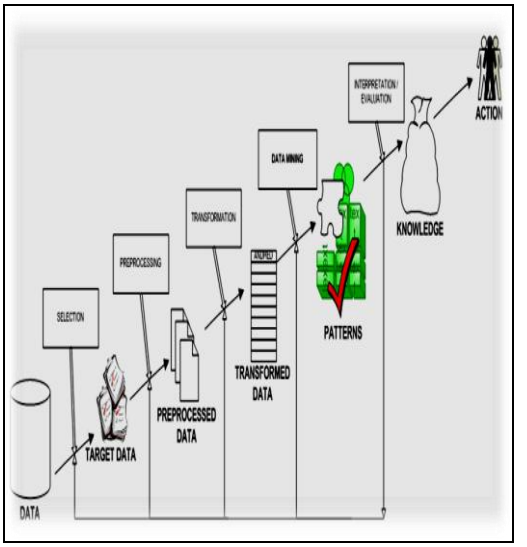
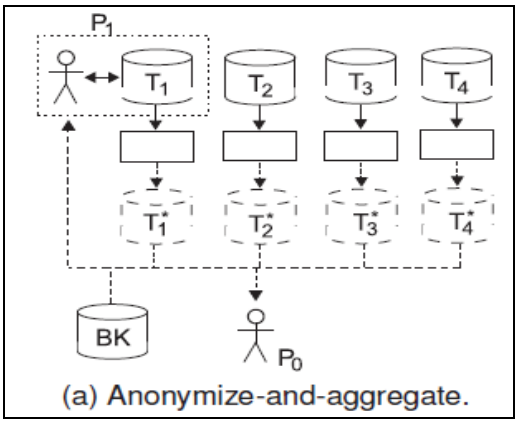


Figure 1.Structure of Data Mining

Privacy preserving data analysis and data publishing **[2]–[4]** have received considerable attention in recent years as promising approaches for sharing data while preserving individual privacy. In a non-interactive model, a data provider (e.g., hospital) publishes a "sanitized" version of the data, simultaneously providing utility for data users (e.g., researchers), and privacy protection for the individuals represented in the data (e.g., patients).

When data are gathered from multiple data providers or data owners, two main settings are used for anonymization [3], [5]. One approach is for each provider to anonymize the data independently (anonymize-and-aggregate, Fig. 2(a)), which results in potential loss of integrated data utility. A more desirable approach is *collaborative data publishing* [3],[5]–[7], which anonymizes data from all providers as if they would come from one source (aggregate-and-anonymize, Fig. 2(b)), using either a trusted third-party (TTP) or Secure Multi-party Computation (SMC) protocols [8], [9].



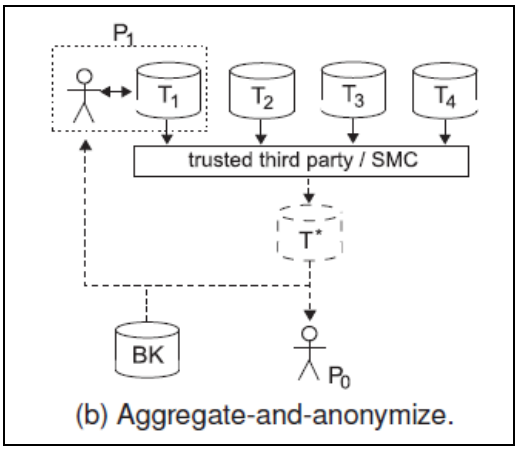


Figure. 2. Distributed data publishing settings for four providers.

1.2 Main Problem

We consider the collaborative data publishing setting (Fig. 2(b)) with horizontally distributed data across multiple data providers, each contributing a subset of records *Ti*. Each record has an owner, whose identity should be protected. Each record attribute is either *an identifier*, which directly identifies the owner, or *a quasiidentifier* (QID), which may identify the owner if joined with a publicly known dataset, or a sensitive attribute, which should be also protected. As a special case, a data provider could be the data owner itself who is contributing its own records. A data recipient may have access to some background knowledge (*BK* in Fig. 2), which represents any publicly available information about released data, e.g., Census datasets.

Our goal is to publish an anonymized view of the integrated data, $T*$, which will be immune to attacks. Attacks are run by *attackers*, i.e., a single or a group (*a coalition*) of external or internal entities that wants to breach privacy of data using background knowledge, as well as anonymized data. Privacy is breached if one learns anything about data

2. Background Knowledge

In this section we will describe the assumptions and background knowledge that is used for developing the new collaborative data publishing.

2.1 Privacy Preserving

The increasing use of data mining tools in both the public and private sectors raises concerns regarding the potentially sensitive nature of much of the data being mined. The utility to be gained from widespread data mining seems to come into direct conflict with an individual's need and right to privacy. Privacy preserving data mining solutions aim at achieving the somewhat paradoxical property of enabling a data mining algorithm to use data *without* ever actually "seeing" it. Thus, the benefits of data mining can be enjoyed, without compromising the privacy of concerned individuals.

For Example

Let $T = \{t1, t2, \ldots\}$ be a set of records with the same attributes gathered from n data providers $P = \{P1, P2, \ldots, Pn\}$, such that $Ti \subseteq T$ are records provided by Pi. Let AS be a sensitive attribute with a domain DS. If the records contain multiple sensitive attributes then, we treat each of them as the sole sensitive attribute, while remaining ones we include to the quasi-identifier **[12].** However, for our scenarios we use an approach, which preserves more utility without sacrificing privacy **[15] .**Our goal, is to publish an anonymized table $T*$ while preventing any m-adversary from inferring AS for any single record. An m-adversary is a coalition of data users with m data providers cooperating to breach privacy of anonymized records.

2.2 *m*-Privacy

To protect data from external recipients with certain background knowledge BK, we assume a given privacy requirement C is defined as a conjunction of privacy constraints:$C1 \wedge C2 \wedge \ldots \wedge Cw$. If a group of anonymized records $T*$ satisfies C, we say $C(T*) = true$. By definition $C(\emptyset)$ is true and $\emptyset$ is private. Any of the existing privacy principles can be used as a component constraint Ci.

We now formally define a notion of m-privacy with respect to a privacy constraint C, to protect the anonymized data against m-adversaries. The notion explicitly models the inherent data knowledge of an m-adversary, the data records they jointly contribute, and requires that each QI group, excluding *any* of those records owned by an $m$adversary, still satisfies C.

2.3 *m*-Privacy with Duplicate Records.

m-Privacy can be also guaranteed when there are duplicate records (such as records from a patient transferred between hospitals). In our initial example Olga has records in two hospitals $P2$ and $P4$ (**Table 1**). For such cases, the duplicates are treated as a single record shared by a few providers. If any of the providers is a member of an m-adversary, the record will be considered as a part of its background knowledge.

TABLE 1

m-Adversary and m-privacy example.

T_1

Name	Age	Zip	Disease
Alice	24	98745	Cancer
Bob	35	12367	Epilepsy
Emily	22	98712	Asthma

T_2

Name	Age	Zip	Disease
Olga	32	98701	Cancer
Mark	37	12389	Flu
John	31	12399	Flu

T_3

Name	Age	Zip	Disease
Sara	20	12300	Epilepsy
Cecilia	39	98708	Flu

T_4

Name	Age	Zip	Disease
Olga	32	98701	Cancer
Frank	33	12388	Asthma

T_a^*

Providers	Name	Age	Zip	Disease
P_1	Alice	[20-30]	*****	Cancer
P_1	Emily	[20-30]	*****	Asthma
P_3	Sara	[20-30]	*****	Epilepsy
P_2	John	[31-34]	*****	Flu
P_2, P_4	Olga	[31-34]	*****	Cancer
P_4	Frank	[31-34]	*****	Asthma
P_1	Bob	[35-40]	*****	Epilepsy
P_2	Mark	[35-40]	*****	Flu
P_3	Cecilia	[35-40]	*****	Flu

T_b^*

Providers	Name	Age	Zip	Disease
P_1	Alice	[20-40]	*****	Cancer
P_2	Mark	[20-40]	*****	Flu
P_3	Sara	[20-40]	*****	Epilepsy
P_1	Emily	[20-40]	987**	Asthma
P_2, P_4	Olga	[20-40]	987**	Cancer
P_3	Cecilia	[20-40]	987**	Flu
P_1	Bob	[20-40]	123**	Epilepsy
P_4	Frank	[20-40]	123**	Asthma
P_2	John	[20-40]	123**	Flu

2.4 Monotonicity of Privacy Constraints

Monotonicity of privacy constraints is defined for a single equivalence group of records, i.e., a group of records that QI attributes share the same generalized values. Let $A1$ be a mechanism that anonymizes a group of records T into a single equivalence group, $T* = A1(T)$. Generalization based monotonicity of privacy constraints has been already defined in the literature **[12], [16].** Its fulfillment is crucial for designing efficient generalization algorithms **[11], [12], [16], [18].** In this paper we will refer to it as *generalization monotonicity*. A privacy constraint C is generalization monotonic if and only if, for any two equivalence groups $A1(T)$

and $A1(T\_)$ that satisfy C, their union satisfies C as well,

$$\begin{matrix} C(\mathcal{A}_1(T)) = true \\ C(\mathcal{A}_1(T')) = true \end{matrix} \Rightarrow C(\mathcal{A}_1(T) \cup \mathcal{A}_1(T')) = true$$

3. Proposed Algorithm

The following is the main anonymization algorithm for m-privacy is introduced in this current paper. We will discuss about that algorithm in detail in the below section.In this section, we present a baseline algorithm, and then our approach that utilizes a data provider-aware algorithm with adaptive verification strategies to ensure high utility and *m*-privacy for anonymized data. We also present an SMC protocol that implements our approach in a distributed environment, while preserving security.

3.1 Anonymization Algorithm

We introduce a simple and general algorithm based on the Binary Space Partitioning (BSP) (Algorithm 1). Similar to the Mondrian algorithm, it recursively chooses an attribute to split data points in the multidimensional domain space until the data cannot be split any further without breaching *m*-privacy w.r.t. *C*. However, the algorithm has three novel features:

Algorithm 1: The *provider-aware* anonymization algorithm.

Data: Records T provided by P_j $(j = 1, \ldots, n)$, QI attributes A_i $(i = 1, \ldots, q)$, the m, and a constraint C
Result: Anonymized T^* that is m-private w.r.t. C
1 π = get_splitting_points_for_attributes (A_i)
2 $\pi = \pi \cup$ get_splitting_point_for_providers (A_0)
3 $\pi' = \{a_i \in \pi, i \in \{0, 1, \ldots, q\} :$ are_both_split_subpartitions_m-private $(T, a_i)\}$
4 **if** π' *is* $\emptyset$ **then**
5 $\quad T^* = T^* \cup \mathcal{A}_1(T)$
6 $\quad$ **return** T^*
7 A_j = choose_splitting_attribute (T, C, π')
8 (T'_r, T'_l) = split (T, A_j)
9 Run recursively for T'_l and T'_r

1. It takes into account the data provider as an additional dimension for splitting;

2. It uses the privacy fitness score as a general scoring metric for selecting the split point;

3. It adapts its *m*-privacy checking strategy for efficient verification. The pseudo code for our *provider-aware* anonymization algorithm is presented in Algorithm 5.

3.2 Secure Anonymization Protocol

Algorithm 1 can be executed in a distributed environment by a TTP or by all providers running an SMC protocol. In this section we present a secure protocol for semi-honest providers. As an SMC schema we use Shamir's secret sharing, but, when needed, we employ also encryption. The key idea of the protocol is to use existing SMC protocols. The first step for all providers is to elect the leader P^I by running a secure election protocol, which then runs Algorithm 2.The most important step of the protocol is to choose an attribute used to split records based on fitness scores of record subsets. Splitting is repeated until no more valid splits can be found, i.e., any further split would return records that violate the privacy.

Secure *m*-privacy anonymization protocol calls three different SMC sub protocols: the secure median ,the secure *m*-privacy verification, and the secure fitness score (Algorithm 3). The last protocol needs to be defined for each privacy constraint *C* (described below). For the sake of this analysis, we assume that all these protocols are perfectly secured, i.e., all intermediate results can be inferred from the protocol outputs.

3.3 Secure Fitness Score Protocol

Many privacy constraints (including ones we have used in our running example) base on threshold values T . In order to securely compare fitness scores of constraints, they need to be *scaled*, e.g., using the least common multiple **(*lcm*)** of all threshold values. After that the secure fitness score can be computed by running the following protocol (Algorithm 3). The elected leader computes the least common multiple of all thresholds from the privacy constraints (line 1). Then, values measured and compared in each privacy constraints can be securely computed (line 3), and *scaled* (line 4).

Shares of the minimal one are scaled back, and returned (line 5).

Algorithm 2. Secure *provider-aware* anonymization protocol

Data: A set of distributed records T, a set of QI attributes A_i $(i = 1, \ldots, q)$, m, a privacy constraint C.
Result: An anonymized view of distributed records $\mathcal{A}(T)$ that is m-private w.r.t. C.

1 $i_{max} = -1$
2 $[f_{max}] = [0]$
3 **foreach** $i \in \{0, \ldots, q\}$ **do**
4 Find the median value s_i of A_i in the set T (using secure median protocol).
5 Send s_i and A_i to other providers.
6 Locally split set T_j into $T_j^{s,i} = \{t \in T_j : t[A_i] < s_i\}$, and $T_j^{g,i} = \{t \in T_j : t[A_i] > s_i\}$.
7 Locally distribute median records among $T_j^{s,i}$ and $T_j^{g,i}$ to reduce uneven distribution of records.
8 Securely verify m-privacy w.r.t. C of a distributed set $T^{s,i} = \bigcup_{j=1}^{n} T_j^{s,i}$ (using Algorithm 2 or 10).
9 **if** $T^{s,i}$ *is not m-private w.r.t.* C **then**
10 **continue**
11 Securely verify m-privacy w.r.t. C of a distributed set $T^{g,i} = \bigcup_{j=1}^{n} T_j^{g,i}$ (using Algorithm 2 or 10).
12 **if** $T^{g,i}$ *is not m-private w.r.t.* C **then**
13 **continue**
14 $[f(T^{s,i})]$ =secure_fitness_score $(T^{s,i})$
15 $[f(T^{g,i})]$ =secure_fitness_score $(T^{g,i})$
16 $[f] = \min([f(T^{s,i})], [f(T^{g,i})])$
17 **if** reconstruct $(lessThan([f_{max}], [f])) == 1$ **then**
18 $[f_{max}] = [f]$
19 $i_{max} = i$
20 **if** $i_{max} \geqslant 0$ **then**
21 Run this protocol for $T^{s,i_{max}}$.
22 Run this protocol for $T^{g,i_{max}}$.

Algorithm 3. Secure Fitness Score Protocol.

Data: $\mathcal{T}$ – thresholds from all constraints, data records T.
Result: Shares of the minimal fitness score value.

1 lcm =least_common_multiple $(\mathcal{T})$
2 **foreach** $i \in \{0, \ldots, w\}$ **do**
3 Securely compute γ_i measured value for C_i, and T
4 $[F_i] = multiplicate([\gamma_i], lcm/\mathcal{T}_i)$
5 **return** $reconstruct(\min([F_1], \ldots, [F_w]))/lcm$

4. Implementation Modules

Implementation is the stage where the theoretical design is automatically converted into practically by dividing this into various modules. We have implemented the current application in Java Programming language with JEE as the main interface for developing the proposed application with Front End as HTML, JSP Pages and Back end as MY SQL data base for storing and retrieving the records. Our proposed application is divided into following 5 modules. They are as follows:

1. Dataset Collection Module
2. Attacks by External Data Recipient Using Anonymized Data
3. Attacks by Data Providers Using Anonymized Data and Their Own Data
4. Doctor Login
5. Secure *m*-Privacy Verification

1. Dataset Collection Module

In this module if patients have to take treatment, he/she should register their details like Name, Age, and Disease they get affected, Email etc. These details are maintained in a Database by the Hospital management. Only Doctors can see all their details. Patient can only see his own record. 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 collaborative data publishing which anonymize data from all Providers as if they would come from one source (aggregate-and-anonymize), using either a trusted third-party(TTP) or Secure Multi-party Computation (SMC) protocols to do computations .

2. Attacks by External Data Recipient Using Anonymized Data

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.

3. Attacks by Data Providers Using Anonymized Data and Their Own Data

Each data provider, such as P1 in Table 1, can also use anonymized data T∗ and his own data (T1) to infer additional information about other records. Compared to the attack by the external recipient in the first attack scenario, each provider has additional data knowledge of their own records, which can help with the attack. This issue can be further worsened when multiple data providers collude with each other.

4. Doctor Login

In this module Doctor can see all the patients details and will get the background knowledge(BK),by the chance he will see horizontally partitioned data of distributed data base of the group of hospitals and can see how many patients are affected without knowing of individual records of the patients and sensitive information about the individuals.

5. Secure *m*-Privacy Verification

In this module Admin acts as Trusted Third Party (TTP).He can see all individual records and their sensitive information among the overall hospital distributed data base. Anonymation can be done by this people. He/She collected information's from various hospitals and grouped into each other and makes them as an anonymized data.

5. Experimental Results

In this paper, we have proposed a new m-privacy method for collaborative data publishing in order to provide security for the sensitive data .For example the patients personal information which is stored in an hospital, it shouldn't be exposed to the third person as that was very sensitive information regarding his case sheet. For this reason only we have examined the proposed paper on behalf of hospital environment by providing privacy for data publishing. We have shown the application in a web interface with JEE 6.0 edition .In this JEE we are using front end as Java Server pages (JSP) and HTML pages. As we are deploying the application in web interface, we are using tomcat server for deploying the application .Hence we use tomcat 7.0 as the deployment web server. We are using My Sql data base for storing the data temporarily on to our system and then retrieve the same data whenever needed.

Main Page

From the above windows we clearly state that this is the home window for the proposed application where the real flow starts from here.It also has the links either to move forward or back ward based on the user opted value.

6. References

[1] N. Li and T. Li, "t-Closeness: Privacy beyond k-anonymity and l-diversity," in *ICDE*, 2007.

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

[3] K. Lefevre, D. J. Dewitt, and R. Ramakrishnan, "Mondrian multidimensional k-anonymity," in *ICDE*, 2006.

[4] C. Dwork, "A firm foundation for private data analysis," *Commun. ACM*, vol. 54, pp. 86–95, January 2011.

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

[6] W. Jiang and C. Clifton, "Privacy-preserving distributed k-anonymity,"in *DBSec*, vol. 3654, 2005, pp. 924–924.

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

[8] O. Goldreich, *Foundations of Cryptography: Volume 2*, 2004.

[9] 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.

[10] P. Samarati, "Protecting respondents' identities in microdata release," *IEEE TKDE*, vol. 13, no. 6, pp. 1010–1027, 2001.

[11] L. Sweeney, "k-Anonymity: a model for protecting privacy," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[12] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-Diversity: Privacy beyond k-anonymity," in *ICDE*, 2006,p. 24.

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

[14] L. Sweeney, "Uniqueness of Simple Demographics in the U.S. Population," Carnegie Mellon University, Tech. Rep., 2000.

[15] T. S. Gal, Z. Chen, and A. Gangopadhyay, "A privacy protection model for patient data with multiple sensitive attributes," *IJISP*, vol. 2, no. 3, pp. 28–44, 2008.

[16] S. Goryczka, L. Xiong, and B. C. M. Fung, "*m*-Privacy for collaborative data publishing," in *Proc. of the 7th Intl. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, 2011.

[17] 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.

[18] B. C. M. Fung, K.Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Comput. Surv.*, vol. 42, pp. 14:1–14:53, June 2010.

7. About the Authors

Kalangi Vilson is currently pursuing his 2 Years M.Tech (CSE) in Department of Computer Science and Engineering at Vignan Institute of Information Technology, Visakhapatnam. His area of interests includes Networks and Data Mining.

Dr.Koduganti Venkata Rao is currently working as Professor and Head of the Department in Department of Computer Science and Engineering at Vignan Institute of Information Technology, Visakhapatnam. His research interests include Security and Cryptography, Parallel Computing & Grid Computing.

