International Journal of Technology and Engineering System (IJTES)
Vol 7. No.1 2015 Pp. 59-63
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-1345

DATA ANONYMIZATION WITH SPECIALIZED RING SIGNATURE TECHNIQUE FOR PRESERVING PRIVACY IN BIG DATA

DEEPALAKSHMI V[1] MAYURANATHAN M[2] T.LAKSHMANAKUMAR[3]
[1,3]PG scholar, [2]Assistant Professor
Department of Computer Science and Engineering
Valliammai Engineering College
Chennai, Tamilnadu-603 203
deepavijay.kpm@gmail.com[a]manimayur@gmail.com[b]laxman2789@gmail.com[c]

ABSTRACT

Sharing of data has been greatly increased nowadays. Almost every individual started sharing their private data's like health record and financial data, etc. For Big data processing framework which rely on cluster computers with a high performance computing platform some parallel programming tools like map-reduce has been used on a large number of computing node. To preserve privacy among the data being shared an effective anonymization technique is used. This paper focuses on describing the anonymization of the data by partitioning the attributes and applying appropriate map-reduce framework on the Hadoop Distributed File System. It also focuses on providing the authorized data and also preserving the identity of the authorized user by making use of an efficient Ring Signature concept.

Keywords: -- Big data, anonymization, Hadoop, MapReduce, sqoop, TaskTracker, JobTracker.

1.INTRODUCTION

Big data is becoming a popular term these days. The term "big" in Big Data changes over time. Every day we send almost 11 billion tests, watches over 2.8 billion you tube videos, perform almost 5 billion Google searches and we are not consuming it instead we are creating it. We create almost 2.5 quintillion bytes data everyday from various sources like social media sites, purchase transaction records, weather reports, cell phone GPS signals, etc. Instead of relying on expensive, proprietary hardware and different systems to store and process data, Hadoop enables distributed parallel processing of huge amounts of data across inexpensive, industry-standard servers that both store and process the data, and can scale without limits. Hadoop can handle all types of data from disparate systems: structured, unstructured, log files, pictures, audio files, communications records, email. Data on individuals and entities are being collected widely. These data can contain information that explicitly identifies the individual (e.g., social security number). Data can also contain other kinds of personal information (e.g., date of birth, zip code, gender) that are potentially identifying when linked with other available data sets. [9,19] Data are often shared for business or legal reasons. This paper addresses the important issue of preserving the anonymity of the individuals or entities during the data dissemination process. The primary focus of this work is to perform anonymization on the data being shared on the public pools that ensures enough privacy on the big data. The major focus of this system lies on the cost effectiveness so that all the people could easily access the data being shared on the public cloud and they can share their own data with the hadoop environment with the full confidence that the sensitive data those they think that are not to be revealed directly to the public can be preserved [12] from others by generalising those data to a certain level. The generalization of data is implemented by specializing or detailing the level of information in a top-down manner until a minimum privacy requirement is violated.

2.RELATED WORK AND PROBLEM ANALYSIS

A.Top down specialization

The generalization of data isusing a specialization or detailed level of information in a top-down manner until a minimum privacy requirement is violated.The privacy goal is given by the*anonymity* on a combination of attributes called a *virtual identifier*,the description on a virtual identifier isrequired to be shared by some minimum number of recordsin the table. A generalization taxonomy tree is specified foreach categorical attribute in a virtual identifier.[1]Map Reduce Top Down Specialization (MRTDS) generalizes the table by *specializing* it

iteratively starting from the most general state. At each step, a general (i.e. parent) value is specialized into a specific (i.e. child) value for a categorical attribute, or an interval is split into two sub-intervals for a continuous attribute. This process is repeated until further specialization leads to a violation of the anonymity requirement.The scale of data in many cloud applications increases tremendously, in accordance with the recent trends in Big Data. The centralized top down specialization approach exploits the taxonomy indexed partition data structure to improve the scalability and efficiency by indexing [11] anonymous data records and retaining statistical information. But in this approach there is an assumption that all data proposed should fit in memory for the centralized approaches. The amount of metadata retained to maintain the statistical information and linkage information is larger.

B.Two phase top down specialisation

Two phase top down approach is to conduct the computationrequired in TDS in a highly scalable and efficient fashion.[6]The two phases are based on the two levelsof parallelization provisioned by MapReduce on cloud. Basically, MapReduce on cloud has two levels of parallelization i.e., job level and task level. Job level parallelization means that multiple MapReduce jobs can be executedsimultaneously to make full use of cloud infrastructureresources. Combined with cloud, MapReduce becomes more powerful and elastic as cloud can offer infrastructureresources on demand, for example, Amazon Elastic MapReduce service [5]. Task level parallelization refersto that multiple mapper/reducer tasks in a MapReduce jobs are executed simultaneously over data splits. It achieveshigh scalability by parallelizing multiple jobs on datapartitions in the first phase, but the resultant anonymization levels are not identical. To obtain finally consistentanonymous data sets, the second phase is necessary tointegrate the intermediate results and further anonymize entire data sets. In the first phase, an original data set D is partitionedinto smaller ones. Thena subroutine is run over each of the partitioneddata sets in parallel to make full use of the job level parallelization of MapReduce. The subroutine is a MapReduce version of centralized TDS (MRTDS) which concretelyconducts the computation required in TPTDS. Two Phase MapReduce Top Down Specialization (TPMRTDS) anonymizes data partitions to generate intermediate anonymization levels. An intermediate anonymization level means that further specialization can be performed withoutviolating k-anonymity. MRTDS only leverages the task level parallelization of MapReduce. In the second phase, all intermediate anonymization levels are merged into one. The basic idea of TPTDS is to gain high scalability by making a tradeoff between scalability and data utility. The slight decrease of data utility can lead to highscalability.

C.Generalized Ring Signatures

The ring signature specifies a set of possible signers instead of revealing the actual identity of the message signer. The verifier can verify that the signature is generated by one of the ring members still he cannot identify which member produced this signature. This can achieve unconditional signer ambiguity and is secure against adaptive chosen-message attacks in the random oracle model. There are certain [7,17]Threshold ring signature enables any group of t entities spontaneously conscripting arbitrary n-t entities to generate a publicly verifiable t-out-of-n threshold signature on behalf of the whole group of the n entities, while the actual signers remain anonymous. [16,18] A highly efficient ID-based ring signature from pairings that requires only one pairing operation is employed. It has the least complexity among its counterparts. AnID-based ring signature (IDRS) scheme has provable security under the standard model and its security fits with the strongest security definition of ring signature.

III. PRELIMINARY KNOWLEDGE ON RING SIGNATURE

A.Id based Threshold ring sign

Threshold ring signature enables any group of t entities spontaneously conscripting arbitrary n-t entities to generate a publicly verifiable t-out-of-n threshold signature on behalf of the whole group of the n entities, while the actual signers remain anonymous. [3,4] Based on the ring signature scheme proposed by HovavShacham and Brent Waters in PKC 2007, an ID-based threshold ring signature scheme can be used for provable security under the standard model and its security fits with the strongest security definition of ring signature proposed by Bender, Katz, and Morselli in TCC 2006.

B.ID-Based Ring Signature from Pairings

A ring signature scheme enables a signer, in an ad hoc manner, to sign a signature on behalf of a group of users including him such that a verifier can be convinced that one of the identified users actually generated the signature but he cannot identify the actual signer. A highly efficient ID-based ring signature from pairings is the one that requires only one pairing operation, [14] which is the least complexity among its counterparts. Additional merits of the scheme include (1) no requirement of admissible encoding function like the Map to Point, and (2) parallelism of computing operations of the ring signature generation.

C.*Identity-based threshold ring signature without pairings*

Identity-based (ID-based) threshold ring signature has rapidly emerged in recent years and many schemes have been proposed until now. However, most of these ID-based threshold ring signatures are constructed from bilinear pairings, a powerful but computationally expensive primitive.[4] Hence, ID-based threshold ring signature without pairing is of great interest in the field of cryptography. An ID-based threshold ring signature scheme without pairings is proven to be existential unforgeable against adaptive chosen message-and-

identity attack under the random oracle model, assuming the hardness of factoring.

D.Attribute based ring sign

An efficient attribute-based ring signature scheme is the one in which the signer signs message by using subset of its attributes. Let all the users with this attributes subset be a ring. It requires that anyone cannot tell who generates the signature in this ring. [2] Furthermore, anyone could not forge the signature for this ring if it not in this ring. It is proved to be unforgeable in the random oracle model and is unconditional anonymous. This new scheme is more efficient and flexible than the previous identify-based ring signature schemes. Compared with the existing attribute-based ring signature schemes, the length of the signature decreases by 1/3, and the pairing operations in these schemes also decrease by 1/3. Thus the efficiency of signing and verifying improves greatly.

E.constant-size signature Bilinear pairing

The ring signature can guarantee the signer's anonymity. Most proposed ring signature schemes have the problem that the size of ring signature depends linearly on the ring size.[13] Some authors have studied the constant-size ring signature to solve the problem. There isan identitybased ring signature scheme with constant size has some security problems by using an insecure accumulator and its verification process does not include the message m. A new scheme with constant-size signature length can be used based on a new secure accumulator from bilinear pairings.

IV. PROPOSED SYSTEM

A scalable two-phase top-down specialization approach to anonymize large-scale data sets using the MapReduce framework has been used. In both phases the approach deliberately designs a group of innovative MapReduce jobs to concretely accomplish the specialization computation in a highly scalable way. The proposed system also focuses on providing the authorized data and also preserving the identity of the authorized user, preserving a highly level privacy on the data as well as the identity of the author.

This system not only focuses on the privacy of the data but also the privacy of the author who owns the data is also preserved using the ring signature. This project is aimed at sharing a data on the public cloud by preserving the sensitive data and also preserving the identity of the author who owns the data. To preserve privacy among the data being shared an effective anonymization technique is used. Data anonymization enables the transfer of information across a boundary, such as between two departments within an agency or between two agencies, while reducing the risk of unintended disclosure, and in certain environments in a manner that enables evaluation and analytics post-anonymization. The proposed system anonymizes the data by partitioning the attributes and applying appropriate map-reduce framework on the Hadoop Distributed File System.

ALGORITHM: DATA PARTITION MAP & REDUCE

MapReduce can take advantage of locality of data, processing data on or near the storage assets to decrease transmission of data.

"Map" step: The master node takes the input, divides it into smaller sub-problems, and distributes them to worker nodes. A worker node may do this again in turn, leading to a multi-level tree structure. The worker node processes the smaller problem, and passes the answer back to its master node.

"Reduce" step: The master node then collects the answers to all the sub-problems and combines them in some way to form the output – the answer to the problem it was originally trying to solve.

Step 1: Input data set D, anonymity parameters k, k^I and the number of partitions p.
Step 2: Partition D into D_i,$1 \leq i \geq p$.
Step 3: Execute MRTDS(D_i,K^I,AL^0) $\rightarrow AL'_i$, $1 \leq i \leq p$ i parallel as multiple MapReduce jobs.
Step 4: Merge all intermediate anonymization levels into one, Merge (AL'_1,AL'_2,...,AL'_p) $\rightarrow AL^I$.
Step 5: Execute MRTDS (D,k,AL^I) $\rightarrow AL^*$ to achieve k-anonymity.
Step 6: Specialize D according to AL^*, Output D^*.

MapReduce is as a 5-step parallel and distributed computation:

Prepare the Map() input – the "MapReduce system" designates Map processors, assigns the K1 input key value each processor would work on, and provides that processor with all the input data associated with that key value.

Run the user-provided Map() code – Map() is run exactly once for each K1 key value, generating output organized by key values K2.

"Shuffle" the Map output to the Reduce processors – the MapReduce system designates Reduce processors, assigns the K2 key value each processor would work on, and provides that processor with all the Map-generated data associated with that key value.

Run the user-provided Reduce() code – Reduce() is run exactly once for each K2 key value produced by the Map step.

Produce the final output – the MapReduce system collects all the Reduce output, and sorts it by K2 to produce the final outcome.

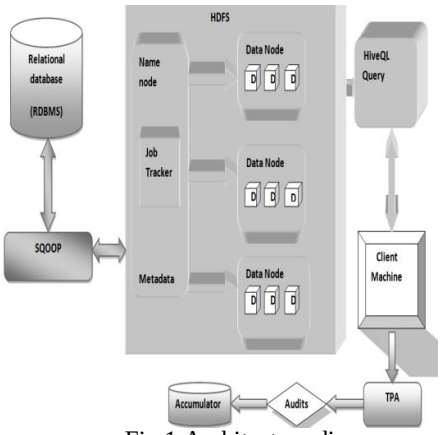

Fig 1 Architecture diagram

In the user interface level the client machine access the data on the hadoop distributed file system through the web browser. In the HDFS the tasks are performed using various nodes. Two important nodes which are to be considered in HDFS are name node and the data node. The name nodes are considered as the job tracker and the data nodes are considered as the task trackers. The JobTracker pushes work out to available TaskTracker nodes in the cluster, striving to keep the work as close to the data as possible. With a rack-aware file system, the JobTracker knows which node contains the data, and which other machines are nearby. If the work cannot be hosted on the actual node where the data resides, priority is given to nodes in the same rack. This reduces network traffic on the main backbone network. If a TaskTracker fails or times out, that part of the job is rescheduled. The TaskTracker on each node spawns off a separate Java Virtual Machine process to prevent the TaskTracker itself from failing if the running job crashes the JVM.The data in the RDBMS are imported to HDFS and exported from HDFS using the tool called sqoop, which is the command line interface application developed by apache hadoop. The third party authority (TPA) will verify whether the authorised person has signed the data and the identities of all those authorised authors will reside in the accumulator. This involves checking whether the person is authorised one or not without revealing the identity of them.

A.MRTDS Driver

Usually, a single MapReduce job is inadequate to accomplish a complex task in many applications. Thus, a group of MapReduce jobs are orchestrated in a driver program to achieve such an objective. MRTDS consists of MRTDS Driver and two types of jobs, i.e., IGPL Initialization and IGPL Update. The driver arranges the execution of jobs. Step 1 initializes the values of information gain and privacy loss for all specializations, which can be done by the job IGPL Initialization.

B.IGPL Initialization Job

The main task of IGPL Initialization is to initialize information gain and privacy loss of all specializations in the initial anonymization level AL.[1] Information gain for a potential specialization in the corresponding Reduce function is computed. The first step is to accumulate the values for each input key. If a key is for computing information gain, then the corresponding statistical information is updated. A salient MapReduce feature that intermediate key-value pairs are sorted in the shuffle phase makes the computation of IG(spec) sequential with respect to the order of specializations arriving at the same reducer. Hence, the reducer just needs to keep statistical information for one specialization at a time, which makes the reduce algorithm highly scalable.

C.IGPL Update Job

The IGPL Update job dominates the scalability and efficiency of MRTDS, since it is executed iteratively, iterative MapReduce jobs have not been well supported by standard MapReduce framework like Hadoop. The IGPL Update job is quite similar to IGPL Initialization, except that it requires less computation and consumes less network bandwidth. Thus, the former is more efficient than the latter.

A.Specialized Ring Signature with one way accumulator

With the help of anonymization we can prevent the private data from being revealed to others unnecessarily. But when sharing data among the group in a public domain, we need to ensure that is data is not being produced by a fake person, hence we use the concept of ring signature to provide authorization.Ring signature, a type of digital signatureis performed by any member of a group of users that each have keys. Therefore, a message signed with a ring signature is endorsed by someone in a particular group of people [18]. One of the security properties of a ring signature is it is computationally infeasible to determine *which* of the group members' keys was used to produce the signature. Ring signatures are similar to group signatures but differ in two key ways: first, there is no way to revoke the anonymity of an individual signature, and second, any group of users can be used as a group without additional setup. [17]A cryptographic accumulatorwhich is a one way membership function is used. [15] It answers a query as to whether a potential candidate is a member of a set without revealing the individual members of the set. One trivial example is how large composite numbers accumulate their prime factors, as it's currently impractical to factor the composite number, but relatively easy to find a product and therefore check if a specific prime is one of the factors. New members may be added or subtracted to the set of factors simply by multiplying or factoring out the number respectively.

V. CONCLUSION AND FUTURE WORK

The very important issues, that is to be concentrated while accessing the data from the public domain is its originality. There are many possibilities in the freely

available public cloud to encounter fake data. Vulnerability in publicly accessible software enables an attacker to puncture the cloud and expose data of other customers using the same service. So considering these issues we focused on ring signature which is similar to the digital signature that can be performed by any member of a group of users that each have keys. Therefore, a message signed with a ring signature is endorsed by someone in a particular group of people. One of the security properties of a ring signature is that it should be computationally infeasible to determine which of the group members' keys was used to produce the signature. Also the work is extended by allowing a third party authority to verify whether the data is authorized or not without revealing the identity of the user.

REFERENCES

[1] *Yeye He, Jeffrey F. Naughton, "Anonymization of SetValued Data via TopDown, Local Generalization", Proceedings of the VLDB Endowment Volume 2 Issue 1, August 2009*

[2] *Wang Wenqiang ; Zhengzhou Inf. Sci. & Technol. Inst., Zhengzhou, China ; Chen Shaozhen, "An Efficient Attribute-Based Ring Signature Scheme", Computer Science-Technology and Applications, 2009. IFCSTA '09. International Forum on (Volume:1), Dec. 2009.*

[3] *Hung-Yu Chien, "Highly Efficient ID-Based Ring Signature from Pairings", Asia-Pacific Services Computing Conference, 2008. APSCC '08. IEEE , Dec. 2008.*

[4] *Xiong Hu, Qin Zhiguang , Li FaGen , JinJing, "Identity-based threshold ring signature without pairings", Communications, Circuits and Systems, 2008. ICCCAS May 2008.*

[5] *Amazon Web Services, "Amazon Elastic Mapreduce," http://aws.amazon.com/elasticmapreduce/, 2013.*

[6] *Xuyun Zhang, Laurence T. Yang, Senior Member, IEEE, Chang Liu, and Jinjun Chen, "A Scalable Two-Phase Top-Down Specialization Approach for Data Anonymization Using MapReduce on Cloud", IEEE Transactions On Parallel And Distributed Systems, Vol. 25, No. 2, February 2014.*

[7] *JianRen, Member, IEEE, and LeinHarn, "Generalized Ring Signatures", IEEE Transactions On Dependable And Secure Computing, Vol. 5, No. 3, July-September 2008.*

[8] *K. Wang, P. Yu, and S. Chakraborty," Bottom-up generalization: a data mining solution to privacy protection", TheFourth IEEE International Conference on Data Mining 2004(ICDM 2004), November 2004.*

[9] *R. Agrawal and S. Ramakrishnan, "Privacy preserving data mining", In Proc. of the ACM SIGMOD Conference on Managementof Data, pages 439–450, Dallas, Texas, May 2000.*

[10] *P.Samarati, "Protecting respondents identities in microdata release", IEEE Transactions on Knowledge and Data Engineering, 13(6):1010–1027,November /December 2001.*

[11] *X.-B. Li and S. Sarkar, "A tree-based data perturbation approach for privacy-preserving data mining ", IEEE Transactions on Knowledge andData Engineering (TKDE), 18(9):1278–1283, 2006.*

[12] *Hongwei Li, Xiao Li, Mingxing He, ShengkeZeng, "Improved ID-based Ring Signature Scheme with Constant-size Signatures", Informatica September 7, 2010.*

[13] *Chengyu Hu , Pengtao Liu, "An enhanced constant-size identity-basedringsignature scheme", Computer Science and Information Technology, ICCSIT, 2nd IEEE International Conference, 2009.*

[14] *ZhiminXu , Tian, Hao , Liu, Dongsheng , Jianming Lin, "A ring-signatureanonymous authentication method based on one-wayaccumulator", Communication Systems, Networks and Applications (ICCSNA),Second International Conference, 2010.*

[15] *Ashish Kumar Kendhe, HimaniAgrawal, "A Survey Report on Various Cryptanalysis Techniques", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-2, May 2013.*

[16] *QuangangZhao,"A New Type of Ring Signature Scheme Based on Group Signatures Idea", Journal of Convergence Information Technology (JCIT), Volume8, issue3.81, Number3, Feb 2013.*

[17] *Jiang Han, XuQiuLiang ; Chen Guohua, "Efficient ID-based Threshold Ring Signature scheme", Embedded and Ubiquitous Computing, EUC '08, IEEE/IFIP International Conference on (Volume:2), Dec. 2008.*

[18] *Xindong Wu, Xingquan Zhu, Gong-Qing Wu, and Wei Ding, "Data Mining with Big Data", IEEE transactions on knowledge and data engineering, vol. 26, no. 1, January 2014.*