Advances in Computer Science and Information Technology (ACSIT)
p-ISSN: 2393-9907; e-ISSN: 2393-9915; Volume 6, Issue 3; July-September, 2019, pp. 139-145

http://www.krishisanskriti.org/Publication.html

Integrated Methodology for Big Data Classification and Security for Improving Cloud Systems Data Mobility

Dr. S. Muthuraj Kumar[1] and E. Nirmala[2]

[1]*Assistant Professor Anna University (Madras Institute of Technology)*
[2]*Scholar Anna University (Madras Institute of Technology)*
E-mail: muthurajkumarss@gmail.com1st,nirmalalochan@gmail.com

Abstract—*The increase in volume and movement of cloud data in the real world, leads to third party threatening. That could be avoided by using certain data protection techniques. Since, most of the cloud data is associated to patient's health information or concerning the personal business. These kinds of data are under high risk, due to malicious access. Conversely, the conventional security solution is competing to handle these issues by means of big data. Due to their quality of mobility and time complexity, high protection is to be provided to the big data. In this article, the proposed architecture incorporates safe and sound security ahead of executing data mobility execution, redundancy and analysis. The security towards data movement can be achieved based on the association between level of risk and its outcome, based on data feature like confidentiality and access mode like public. The Hadoop Distributed File system is used towards big data, based on the aforementioned classification. This, predicts from malicious access, and ensures security to data mobility in cloud environment.*

1. INTRODUCTION

The growing volume of data due to social media, business, finance and healthcare, needs a huge storage and retrieval facility with a reduced amount of risk. This has its own process to analyze and correlate the user needed information. Distributed File handling technique called Hadoop File Handling System (HDFS) acts as a tool, used to reduce redundant data, to implement effective data storage and mining. Whenever, the Big Data is used to handle variety of data like structured, semi structured and unstructured, Hadoop File System (HDFS) on the distributed environment helps to store and administer these data sets to avoid redundant data file, and improve the automated decision making process in critical situation to reduce risk factor. This always ropes to loyalty, dependability, elasticity and similar dispensation in distributed structural design, then fog computing came into picture to improve the security by storing the data at different servers with different ratios, by restricting the hackers from malicious access. This can be achieved by using the technology called big data with data mobility, by using Map Reducing Scheme, to improve the effective utilization of the storage space.

The goal of the cyber hackers is to attack the cloud data via virtual communication. Guest operating system generates periodically the log table entry of virtualized machines. By collecting such information, the attack type is extracted, by making use of the Map Reduce technique (MR). The parser in the MR is used to identify the nature of the attack, and the percentage is calculated through Machine Learning and logical degeneration technique.

The presence of the attack is observed by applying malware with existing virtualized schemes. This is very cost effective with minimum operational overhead. In conventional cloud technology outsourcing of data leads to security leakage.

This has been ignored in the past. But could be resolved using a methodology called linear System Outsourcing (LSO) is having high security without extricating spsaring This reduces the complexity in calculation by solving linear regressive equations. These sort of linear calculations helps to reduce the input output complications with respect to memory and also increases the protection level at user level. The calculations has been implemented on to the Amazon Web Services (AWS) and proven with efficiency with security up to 74%. on big data in cloud environment.

Memory Management in Big data has a new technology with as a mile stone called MemepiC. This has huge no of users, due to attracted feature called DRAM allocation along with advanced hardware devices. On the other hand this has a greater progression in the systems functioning with respect to its memory. In case of data analytics and storage MempiC helps to perform the task with low latency with increased efficiency in storage of data depending up on its analytics. Through this technology, Remote Direct Memory Access (RDMAS) scheme is used. In Big Data, data analytics plays a imperative role in decision making process. This is achieved by sending the key value of distributed storage.

2. LITERATURE SURVEY:

Author Ke Cheng et.al. [7] Proposed a system for spatial big data analytics using secured cloud environment. In his methodology author uses multimedia data bases to provide the services using clustering. The advantage is it ensures secrecy and privacy to handle the encryption and decryptions separately. While the data is outsourced, Here, it is the

decision of the owner to encrypt and decrypt by using his own keys or multiple keys without any dependency and sharing. This enhances both secrecy and confidentiality to Big Data.

Jun Wu, et.al. [8] Presented a technique based on sophisticated connections through interaction to provide better security to huge user information called big data. This stimulates providing security by detecting the issues primarily to increase the confidentiality. Author addresses the problem, and proposes a new line of attack, called situational awareness mechanism to predict the security issues in any type of network associated to big data and smart grid

B. Saraladevia et al. [9] proposes a system based on big data through Hadoop File Systems, to direct, allocate and save multi user information. The author states different problems with security perspective in Hadoop Design architecture in HDFS. And also author ensures data security by following three different methodologies like HDFS, Kerberos, and algorithmic approach. These techniques were used on the sensitive data to reduce the redundancy of user data. So, finally, author concludes that integration of these three technologies has improved security.

Thu Yein Win et.al.[10] proposed a system to enhance the security in the virtualized environment of big data cloud Architecture. In real time, author addressed the problem like cyber attackers, to prevent from such attacks author introduced a mechanism called mining of attack based on identification and attacking path. These are created by using chart based occurrence association used to identify the exact user by using fuzzy identification logic; chance of attack is calculated based on key fields of the attacker, to determine the type and no of occurrence of the attack. So, this technique could be used to detect advanced attacks in VM resided in Big Data using HDFS Map Reduce Approach.

Zheng Yan,et.al.[12] proposes a new technique to detect the data redundancy. So far, there is no technique to detect redundancy in the encrypted file. The proposed technique will work on the basis of user encrypted data after deducting duplicate entries, by applying a mechanism called as proxy reentry. The advantage of this technique is its efficiency in storage and retrieval with reduced time. The proposed system with encrypted deduplication supports for updations at user level over distributed data, even the owners of database are offline.

Maziar Goudarzi et.al.[13] proposed a system heterogeneous data processing for secured big data comparatively gives better processing results. And the conventional system is working on the basis of batch process oriented applications. This envisioning technology gave a path on extracting and needed Information by gaining knowledge using the Map Reduce (MR) architecture type. And also author addressed on the subject of various challenging feature extraction technique with data analytics and in future the research is predicted to extend batch processing with the forthcoming trend related big data analytics and data processing.

Sergio Salinas et.al.[14] proposed a system to solve the sparse linear systems of equations in big data. In the proposed system author notified that Linear System of equations (SLSE) are used to hide the security related legacy information against the service providers due to security policy. Compared to the existing system with LSE. The proposed system has been proven for its execution against user security, with less complexity and reduced error during the process of outsourcing. matrix technique is an added advantage, Implemented in Amazon Elastic Cloud EC2 and proven the efficiency in performance at 74%.. This has proven the privacy with SLSE, but in future running time can be reduced.

3. SYSTEM ARCHITECTURE

The architectural design of HDFS in Fig.1 states the file system with file loading, the different structures of files are generated every second in terabytes and petabytes as in the form of audio, video, image and text. And may be structured, unstructured or semi structured in distributed environment of cloud architecture. The conventional big data is suffering from security due to public storage access. So the proposed design of big data has been created with map reduce MR-Technique to handle sensitive data in mission critical applications. There are six different phases in our proposed design aim to address the methodology to incorporate secrecy related procedure to reduce the risk of data movement in cloud, by constructing the data security through algorithmic approach, by following the design constructs as follows.

HDFS-File System Initially it is constructed to provide security to the big data. But it is hard to implement for the huge volume of data. The data delivered to the end user is in the form of key value. Securing the whole data or data set is difficult in real time instead attribute or field specified by the user can be protected. The factors which determine the construction of classification depends upon categorization, forecasting, grouping and relationship rule discovery.

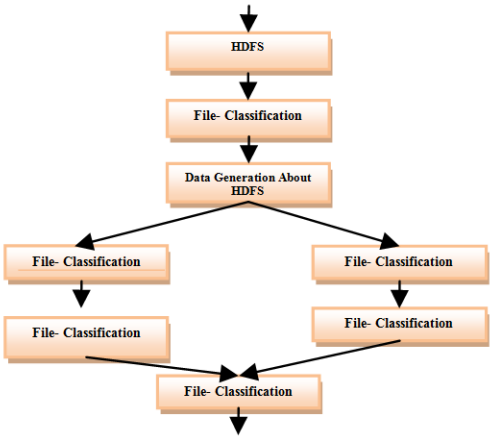


Fig. 1: HDFS Architecture

Based on these four factors the classification architecture is constructed with the following phases.

HDFS-File Processing Phase

The design structure starts with the first phase called Hadoop File System (HDFS) these files may consists the text data, video, and audio. This is to be processed in the first phase. Further the unstructured or semi structured data has to be sent to processing.

Data Generation regarding the File Content Phase

In this phase the data on the file, like the name, no of attributes, size, time and owner are generated. If the data concerning the file is not generated along with the file in transfer, immediately the relevant information is inserted to the file at this level. Then classification is carried out based on the request. If the file is public, then no additional field is added towards the security of the data. If not additional field for security will be added.

HDFS-Classification Phase Data with reference to the file is to be generated to group the file content under distributed environment. Classification technique is proposed to carry out the task of generating the data in relation to file content like the name with extension, time stamp, size etc. In this situation the different format of files are classified and segmented into different no of logical spells. Each file may be of equal in size like 128MB. Then, each such reduced spells of data is sent to the map reduce Phase.

HDFS-Mapping Phase The spells of data size of 128 MB are checked for its availability with respect to the memory. If the existing data size is matches with the memory, then the data size is send to the mapping process. If not, based on the classification file data is checked for its mapping process with one or more Mapper, by converting their different format into the unique text file. A constraint on the nature of the data like private of public data is confined before mapping process by verifying the attributes like data on the subject of the file.

HDFS-Reducing Phase In this phase the classified data set is checked for its redundancy. The output from the mapper process is passed as an input. If any such redundant data is found while storing, transferring and loading process, this may cause more storage space and unauthorized access with data leakage. To overcome with this problem, redundant data set is deleted from the memory for effective utilization. Thereby, the Reducing phase generates the output zero for public data, one for the private data with secrecy.

HDFS-Security

The user data along with his reference to file content has been send to the cloud are encrypted, with allocated key certificates, to enhance security to data in transit. By making use of the random key generated by the sender, data transmission is taken place by referring the block identifier with the name node. After receiving the identified blocks, the receiver generates the sharable key to the sender based on the name node. Then, the data node of the receiving end uses the encrypted key to activate a request towards migration of data. After receiving any such request the sender started decrypting the data for authorization, by getting the approval for the packet of information starts transmitted from the sender to the receiver through response. The process is repetitively continued till confirmation is generated, to check for its availability. Whenever, the process of transmission ends immediately hash value for the data is generated to check for its timestamp. This is how the process of communication is implemented

HDFS-Algorithms

The HDFS algorithm has been segmented into two major parts, similar to data categorization and security enhancement. As soon as files are loaded in the cloud, the classification algorithm classifies the data files based on the type and splits the file content into similar blocks of 128 Mb. After classification security is enhanced with that key value of the data next to file content. Data in relation to the file content is attached for the private file. If it is public data then simply file content is spitted into no of similar block size with 128MB.

Algorithm for Categorization

Step1: Start HDFS-File processing

Step2: Obtain the data about the file content DAFC Of HDFS.

Step3: If data about file content is found then

Step4: goto Security Enhancement Algorithm SEA

Step5: else

Step6: if type of the file is not text then convert into text file then

Step7: do steps from 12-15

Step8: else

Step9: do steps 9 -11

Step10: HDFS-Data about file content is to be defined DAFC

Step11: Put the value for HDFS- DAFC

Step12: go to SEA

Step13: Conversion of file into text format

Step14: Implementing HDFS-Mapping process to each data segment

Step15: Determine the HDFS is private or public based on the segment of whole file content.

Step 16: Obtain the results of the HDFS-Classifier

Step 17: HDFS-Reducer results into a single unit of data by avoiding redundancy.

Step 18: If the HDFS-Mapping is confidential then

Step 19: Then result is checked for its privacy DAFC of HDFS.

Step 20: Else

Step 21: Then the result obtained is public

Step22: If the result obtained by HDFS-Reducer is private

Step23: Then result is checked for its privacy DAFC of HDFS

Step 24: Goto SEA

Step 25: Else

Step 26: Then the result obtained is public

Step 27: Goto SEA

Step 28: Stop Processing.

The above algorithm works on to the loaded files into HDFS for classification, then security is enhanced with encryption by getting the consent from both the parties for the data block then, public files are stored as such by secret files are encoded with secret key value using HDFS-SEA, then sent calling HDFS-Mapping and HDFS-Reducing process.

Security Enhancement Algorithm:

Step1: Start HDFS-SEA

Step2: if the data from HDFS- DAFC is secured then

Step3: do steps 5-23

Step4: else

Step5: goto step 23

Step6: Cloud sender Sends Name Node to SCN sends DAFC with shared keyvalue p_{key} to Cloud Receiver Name Node.

Step7: Destination Cloud Data Node sends tokens to access the blocks of data P using p_{key} to Cloud Receiver and request for transferring the file content.

Step8: Data Node of the Cloud sender verifies the authentication based on key value p_{key}

Step9: After receiving the p_{key} the cloud sender Sends the received token along with the p_{key} with its corresponding hash value to the receiver cloud.

Step10: Cloud Receiver checks the p_{key} with its hash value send by the Cloud Sender

Step11:Cloud Receiver transmits the acknowledgement along with p_{key} to Cloud Receiver

Step12: if time of transaction is less than 0

Step13: Cloud Senders used to wait for the acknowledgement from the receiver.

Step14: else

Step15: if the transaction is retransmitted due to denial of service

Step16: repeat the above stated step 9 with p_{key} and its hash value

Step17: else

Step18: Cloud Sender's Data Node is impelled

Step19: if the redundant copy of the Receiving cloud is greater than the maximum value set then

Step20: The Receiving Cloud is Data is provoked

Step21: The Cloud Sender will receive the acknowledgement and check for the data or file content with Data Node

Step22.: End

Step23: End HDFS-SEA

4. Performance Analysis The efficiency of the algorithm is measured based on the factor called Name Node of the Cloud Data Sender and Cloud Data Receiver. Here the key generation is done by the Cloud Data Receiver with random key with hash value. Based on the Hash Table Entry tokens are generated to create two valued hash entries to encrypt without any delay to transform the data content. By implementing this conventional technique, time and cost overheads are huge. to overcome with this complexity cloud data sender and cloud data receiver among acknowledgement phase is implemented so that packets can be transmitted with reduced time and cost overhead. So that HDFS-SEA algorithm enforces security to data in transit via encryption and decryption to those of private in nature as in the form of key generation and acceptance technique. this is highly confidential and enhances the security aspect in HDFS via classification practice. HDFS-Map Reducing technique allows to avoid the redundant data improves the effective utilization of storage efficiency.

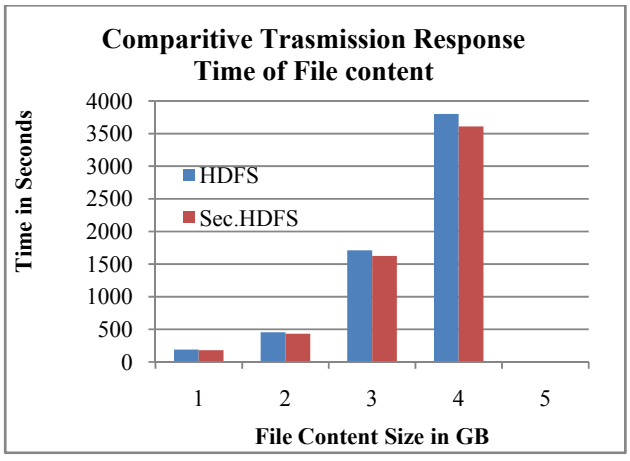
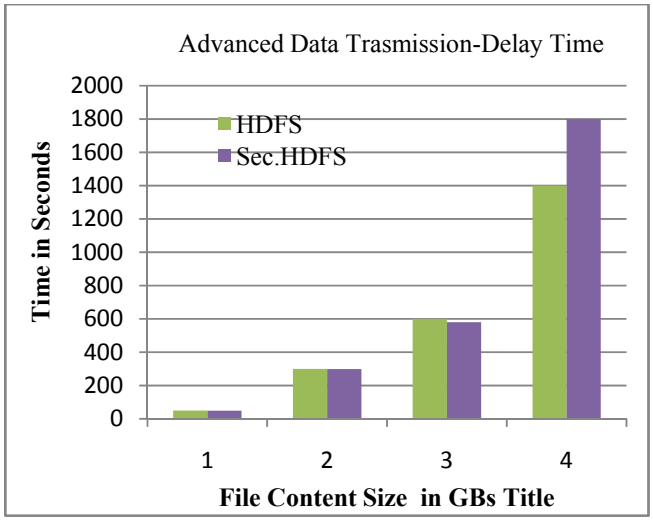


Fig. 2: Comparative Transmission Response Time of File content



5. Conclusion The advanced security in big data could be achieved by the aforementioned algorithms like HDFS-Classification and HDFS-SEA. This exhibits a challenge in reduced risk with high efficiency in providing security towards data transmitted in the cloud . The key invention technique helped to enhance the security by verification of the user and receiver through confidential acknowledgement process. The HDFS-Mapping technique maps the corresponding file content from the source to destination, irrespective of the data type and content. The HDFS-Reducing functionality helps in achieving the effective storage utilization by deleting the redundant content. In future, the time delay is intended to be reduced again to transfer more content in Big Data by reducing the delay time.

References

[1] Thu Yein Win , Huaglory Tianfield, and Quentin Mair",Big Data Based Security Analytics for protecting Virtualized Infrastructures in Cloud Computing",IEEE Transactions on Big Data, 4,1, March 2018,pp.11-25.

[2] Jun Wu , Kaoru Ota , Mianxiong Dong , Jianhua Li, and Hongkai Wang," Big Data Analysis-Based Security Situational Awareness for Smart Grid",*IEEE Transactions on Big Data*,4,3,September 2018.pp.408-418.

[3]Sergio Salinas , Member, Changqing Luo ,Xuhui Chen, Weixian Liao,Pan Li ,"Efficient Secure Outsourcing of Large-Scale Sparse Linear Systems of Equations", *IEEE Transactions on Big Data*,4, 1, March 2018, pp.26-38.

[4] B. Saraladevia , N. Pazhanirajaa , P. Victer Paula , M.S. Saleem Bashab , P. Dhavachelvanc, "Big Data and Hadoop-A Study in Security", *Perspective Procedia Computer Science*, 50,2015, pp.596-601.

[5] Xiao HongJu, Wang Fei, Wang FenMei, Wang XiuZhen, "Some Key Problems of Data Management in Army Data Engineering Based on Big Data", IEEE *2nd International Conference on Big Data Analysis*, October 2017.pp.149-151.

[6] Qingchao Cai , Hao Zhang , Wentian Guo, Gang Chen, Beng Chin Ooi , Kian-Lee Tan,and Weng-Fai Wong, "MemepiC towards a Unified In-Memory Big Data Management System", *IEEE Transactions on Big Data,* 5,1 March 2019. pp.4-17.

[7] Ke Cheng, Liangmin Wang,Yulong Shen, Hua Wang, Yongzhi Wang, Xiaohong Jiang and Hong Zhong, " Secure k-NN Query on Encrypted Cloud Data with Multiple Key", IEEE Transactions on Big Data Journal of Latex Class Files..4,8, Early access 2017.pp1- 1

[8] Jun Wu , Kaoru Ota , Mianxiong Dong ,Jianhua Li, and Hongkai Wang, "Big Data Analysis-Based Security Situational Awareness for Smart Grid", *IEEE Transactions On Big Data* ,4, 3, September 2018pp.408-417.

[9] B. Saraladevia, N. Pazhanirajaa, P. Victer Paula, M.S. Saleem Bashab, P. Dhavachelvanc" Big Data and Hadoop-*A Study in Security perspective, 2nd International Symposium on Big Data and Cloud Computing (ISBCC'15*) ",Procedia Computer Science 50,2015,pp. 596 – 601,2015.

[10]. Thu Yein Win , Huaglory Tianfield, and Quentin Mair, "Big Data Based Security Analytics for Protecting Virtualized Infrastructures in Cloud Computing", *IEEE Transactions on Big Data.4,*1, March 2018. pp.11-25.

[11]. Aniello Castiglione, Giuseppe Cattaneo, Giancarlo De Maio, Alfredo Deantis,and Gianluca Roscigno," A Novel Methodology to Acquire Live Big Data Evidence from the Cloud", ***IEEE Transactions On Big Data, Early Access,*** March ***2017***,pp.1-14.

[12] Zeng Yan, Senior Member, IEEE, "Wenxiu Ding, Xixun Yu, Haiqi Zhu, and Robert H. Deng,"Deduplication on Encrypted Big Data in Cloud",*IEEE Transactions On Big Data*,2,2, June 2016 pp.138-150.

[13] Maziar Goudarzi , Senior Member, "Heterogeneous Architectures for Big Data Batch Processing in MapReduce Paradigm", ***IEEE Transactions On Big Data,5***,1, March 2019, pp.18-33.

[14] Sergio Salinas , Changqing Luo , Xuhui Chen,Weixian Liao ,Pan Li , " Efficient Secure Outsourcing of Large-Scale Sparse Linear Systems of Equations", *IEEE Transactions on Big Data,* 4,1, March 2018, pp26-39,.

[15] S.H. Kim, N.-U. Kim, and T.-M. Chung,`Attribute relationship evaluation methodology for big data security," *In Proceedings of International Conference on IT Convergence Security (ICITCS),* January 2013, pp. 1-4.

[18] Q. Shen, L. Zhang, X. Yang, Y. Yang, Z. Wu, and Y. Zhang, ``SecDM: Securing data migration between cloud storage systems, "I***n Proc. IEEE 9th Int. Conf. Dependable, Autonomic Secure Computing*** (DASC), December. 2011, pp. 636-641.

[19] Ismail Hababeh, Ammar Gharaibeh, Samer Nofal and Issa khalil ," An Integrated Methodology for Big Data Classification and Security for Improving Cloud Systems Data Mobility",*IEEE Access, December 2018*, 7,pp.9153 – 9163.

[20] Marwa Elsayed and Mohammad Zulkernine," Towards Security Monitoring for Cloud Analytic Applications", *4th IEEE International Conference on Big Data Security on Cloud* November 2018,pp.69-78.

All printed material, including text, illustrations, and charts, must be kept within a print area of 6-7/8 inches (17.5 cm) wide by 8-7/8 inches (22.54 cm) high. Do not write or print anything outside the print area. All text must be in a two-column format. Columns are to be 3-1/4 inches (8.25 cm) wide, with a 5/16 inch (0.8 cm) space between them.

Text must be fully justified. A format sheet with the margins and placement guides is available in Word files as <format.doc>. It contains lines and boxes showing the margins and print areas. If you hold it and your printed page up to the light, you can easily check your margins to see if your print area fits within the space allowed.

4. MAIN TITLE

The main title (on the first page) should begin 1-3/8 inches (3.49 cm) from the top edge of the page, centered, and in Times 14-point, boldface type. Capitalize the first letter of nouns, pronouns, verbs, adjectives, and adverbs; do not capitalize articles, coordinate conjunctions, or prepositions (unless the title begins with such a word). Leave two blank lines after the title.

5. AUTHOR NAME(S) AND AFFILIATION(S)

Author names and affiliations are to be centered beneath the title and printed in Times 12-point, non-boldface type. Multiple authors may be shown in a two- or three-column format, with their affiliations below their respective names. Affiliations are centered below each author name, italicized, not bold. Include e-mail addresses if possible. Follow the author information by two blank lines before main text.

6. SECOND AND FOLLOWING PAGES

The second and following pages should begin 1.0 inch (2.54 cm) from the top edge. On all pages, the bottom margin should be 1-1/8 inches (2.86 cm) from the bottom edge of the page for 8.5 x 11-inch paper; for A4 paper, approximately 1-5/8 inches (4.13 cm) from the bottom edge of the page.

7. TYPE-STYLE AND FONTS

Wherever Times is specified, Times Roman, or New Times Roman may be used. If neither is available on your word processor, please use the font closest in appearance to Times that you have access to. Please avoid using bitmapped fonts if possible. True-Type 1 fonts are preferred.

8. MAIN TEXT

Type your main text in 10-point Times, single-spaced. Do not use double-spacing. All paragraphs should be indented 1 pica (approximately 1/6- or 0.17-inch or 0.422 cm). Be sure your text is fully justified—that is, flush left and flush right. Please do not place any additional blank lines between paragraphs. You can use courier for source code.

For I = 1 to 10 do

Print “this figure”

End;

Figure 2: Example figure.

8.1 Figure and Table Captions

Captions should be 9-point Times New Roman font, boldface. Callouts should be Times New Roman, non-boldface. Initially capitalize only the first word of each figure caption and table title. Figures and tables must be numbered separately. For example: “Figure 2: Example figure.”, “Table 1. Table example.”. Figure captions are to be **below** the figures (see Figure 2). Table titles are to be centered **above** the tables (see Table 1.

Table 1. Table example.

One	Two	Three

9. FIRST-ORDER HEADINGS

For example, “1. Introduction”, should be Times 12- point boldface, initially capitalized, flush left, with one blank line before, and one blank line after. Use a period (“.”) after the heading number, not a colon.

9.1 Second-order Headings

As in this heading, they should be Times 11-point boldface, initially capitalized, flush left, with one blank line before, and one after.

9.1.1 Third-order Headings. *Third-order headings, as in this paragraph, are discouraged. However, if you must use them, use 10-point Times, boldface, initially capitalized, flush left, preceded by one blank line, fol lowed by a period and your text on the same line.*

10. ACKNOWLEDGEMENTS

This work was supported in part by a grant from the National Science Foundation.

REFERENCES

List and number all bibliographical references in 9- point Times, single-spaced, at the end of your paper. When referenced in the text, enclose the citation number in square brackets, for example [2-4], [2, 5], and [1].

[1] Briand, L. C., Daly, J., and Wüst, J., "A unified framework for coupling measurement in objectoriented systems", *IEEE Transactions on Software Engineering*, 25, 1, January 1999, pp. 91-121.

[2] Maletic, J. I., Collard, M. L., and Marcus, A., "Source Code Files as Structured Documents", in *Proceedings 10th IEEE International Workshop on Program Comprehension (IWPC'02)*, Paris, France, June 27-29 2002, pp. 289-292.

[3] Marcus, A., *Semantic Driven Program Analysis*, Kent State University, Kent, OH, USA, Doctoral Thesis, 2003.

[4] Marcus, A. and Maletic, J. I., "Recovering Documentation-to-Source-Code Traceability Links using Latent Semantic Indexing", in *Proceedings 25th IEEE/ACM International Conference on Software Engineering (ICSE'03)*, Portland, OR, May 3-10 2003, pp. 125-137.

[5] Salton, G., *Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer*, Addison-Wesley, 1989.