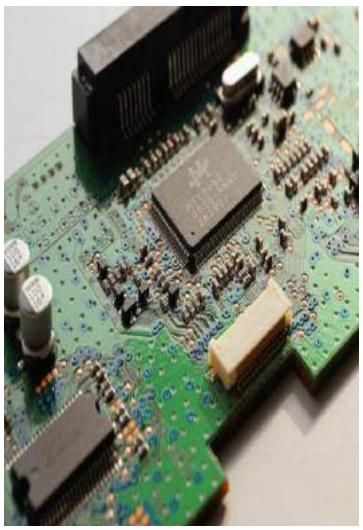
ISBN: 978-0-9957075-1-1

INTERNATIONAL JOURNAL of ENGINEERING AND APPLIED COMPUTER SCIENCE

EMPIRICAL RESEARCH PRESS

Volume: 01 | Issue: 02 | December | 2016






EMPIRICAL RESEARCH PRESS LTD.

Kemp House, 160 City Road, London
United Kingdom

IJEACS

International Journal of

Engineering and Applied Computer Science

Empirical Research Press Ltd.

London, United Kingdom

Volume: 01, Issue: 02

ISBN: 978-0-9957075-1-1

www.ijeacs.com

Indexing, Hosting and Advertising

Internet Archive

Message

International Journal of Engineering and Applied Computer Science (IJEACS) is an open access, double-blind peer reviewed international journal, monthly publishing online by Empirical Research Press Ltd. Empirical Research Press is a research publishing company with name, trademark registered, incorporated in England and Wales, United Kingdom.

The scope of International Journal of Engineering and Applied Computer Science is to publish high quality research contributions, latest innovations, advance development carried out in the field of Engineering, Computer Science and Technology. The original research, review, case study, survey, new interpretation and implementation of concepts and theories, recent technological development, technical reports, empirical discussions are invited to submit for publication.

The major objectives of International Journal of Engineering and Applied Computer Science are to maintain high quality of publications and indexing with world's highly accessed and cited research and academic databases. The scope of IJEACS publications also includes special and interdisciplinary research contributions. We offer best wishes to readers, reviewers and contributors of IJEACS.

Board Members of IJEACS

Content

International Journal of Engineering and Applied Computer Science (IJEACS)

Features for Detecting Malware on Computing Environments

Ajit Kumar, K.S. Kuppusamy
Department of Computer Science
Pondicherry University
Puducherry-605014, India

G. Aghila
Department of Computer Science and Engineering
NIT Puducherry
Karaikal-609609, India

***Abstract*—Malware is the main threat for all computing environments. It also acts as launching platform for many other cyber threats. Traditional malware detection system is not able to detect "modern", "unknown" and "zero-day" malware. Recent developments in computing hardware and machine learning techniques have emerged as alternative solution for malware detection. The efficiency of any machine learning algorithm depends on the features extracted from the dataset. Various types of features are extracted and being researched with machine learning approach to detect malware that are targeted towards computing environments. In this work we have organized and summarized different feature types used to detect malware. This work will direct future researchers and industry to make decision on feature type selection regarding chosen computing environment for building an accurate malware classifier.**

Keywords- malware; computing environment; cyber-threat; feature type; machine learning

I. INTRODUCTION

Malware is a computer program, intensely written to harm computing resources. Malware are of different type based on structural and behaviors difference, such as Virus, Worm, Trojan, Bot, Spyware, Adware, Rootkit, Bootkit, Ransomware etc. Growth of variant of known malware and new malware is increasing year-by-year [1] and posing threat to digital infrastructure.

Malware is main threat for all four kind of computing environments: 1) Personal computing; 2) Mobile computing; 3) Embedded computing; and 4) Industrial control system (ICS) computing. Although a large percentage of total malware targeted for first two environments i.e. personal and mobile computing, recent past have seen a major surge in malware targeting other two environments as well i.e. embedded and ICS computing. To tackle malware threats personal and mobile computing have some traditional solutions such as signature and heuristic based anti-malware but embedded and ICS computing are wide open for malware attack. Traditional solutions are not effective in detecting malware at either of computing environments as they have inherited limitations [2]–[4].

Signature based techniques are backbone of these traditional anti-malware solutions which itself is totally unable to work against *"modern", "unknown" and "zero-day" malware*. Signature based techniques works in two phases: 1) Signature creation; and 2) Signature matching. Signature creation is a multi-step process which involves steps such as malware collection, malware analysis, signature generation and signature distribution. All of these steps are carry out with the help of human and machine in proportion. Human involvement makes process costly and slow which provide a large attack window to malware. Machine works on the principal of generalization which misses artifacts of "modern", "unknown" and "zero-day" malware. Signature matching is also mutli-step process such as file scanning, signature look-up and alerting user. It performance is dependent on previous phase i.e. signature creation because it can only match signatures which are in the database and so "zero-day" and "unknown" malware will escape the detection. Apart from technical bottlenecks, signature based techniques also suffers with other drawbacks like costly analysis process, high computation and memory requirements at end host and requirement of regular signature updates. These bottlenecks and drawbacks of signature based techniques created a need of alternative anti-malware solution and machine learning based techniques is emerging to fulfill the same.

Machine learning techniques consider malware detection as a binary or multi-class classification problem which is similar to many other domains. Machine learning based malware detection has two phases: 1) training; and 2) classification. Training is a multi-step process and sequential in nature. Steps involve in building malware classifier are: sample collection, sample labeling, feature extraction, feature selection and model building. Sample collection is process of collecting malware and benign programs which is precedence by sample labeling which assign true class label (malware and benign) to each sample. Labeled samples are ready for further step which is feature extraction. Feature extraction is very important step of overall machine learning process. Extracted features mainly decide classifier performance and so with different type of features classifier perform differently. This decisive nature of feature attracts lots of engineering methods to extract different

ISBN: 978-0-9957075-1-1

type of features which have more discriminative values than others. Feature selection deals with excessive extracted features and help to filter out only few useful features on the basis of discriminative rank. Model building is last step of training, it takes selected features and run machine learning algorithms which output a model which would be able to classify inputted new sample. During classification phase each sample goes through the feature extraction phase but now only those features are extracted on which the model is built. Built model takes these extracted features as input and output the probable class label for each sample.

Machine learning based malware detection is suitable for "modern", "unknown" and "zero-day" malware detection because it is not per sample base technique as signature based techniques are, instead it works by learning malware and benign classification based on extracted features from training dataset and able to generalized to unseen samples. Features type plays an important and decisive role for accurate malware detection using machine learning. Many researches in domain of malware detection using machine learning focus on different feature type which are extracted by various methods and impact classifier performance.

Over years many feature type for malware detection is proposed and experimented which are spread over all of computing environments. In this work, we have organized and summarized different feature type used for various file types on different computing environments. Due to vary computing architecture, supported file types and analysis method among different computing environments, feature types vary across these environments. Classifiers performance depend largely on features types and feature selection, hence having a well organized literature on feature types will help in easy decision making for various entities involved such as future researchers and industries developers.

II. METHODS

Feature type can be group primarily according to computing environments and further under each computing environment it can be organized according to the analysis type. In this section four computing environments are explained and then each of analysis type is explained.

A. *Computing Environments*

Computing environment is term use to describe a complete computing platform comprise of hardware, operating system (OS), and other software. Each computing environments differs on aforementioned components. Each of computing environment is explained further.

1)Personal computing: Personal computing refer to the use of Desktop and laptop computer which are used in normal day-to-day life and in various enterprises to automate the tasks. Distinguished dimensions of Personal computing environment are a bigger output screen, high internal and external memory, desktop OS and attached keyboard and mouse.

2)Mobile computing: Mobile computing refer to all of those devices which are mobile in nature and having a smaller screen size than personal computing devices and limited with small battery. All such devices have specific mobile operating system and customized operating system. Android, iOS and Windows are there main leading mobile OS.

3)Embedded computing: Embedded computing refers to all those smart digital devices which have configuration options and run a specialized operating system designed for such embedded system. OS running in smart car, smart home, modern freeze and many other modern digital appliances are example of embedded computing.

4)ICS computing: Industrial control system (ICS) computing refers to those devices which run specialized OS and software to control and monitor industrial system such as digital power and water distribution system, nuclear plant etc.

B. *Analysis and Feature Type*

Feature extraction involves two type of analysis which carried out in different manners and gives features which have vary discriminative values. Static and dynamic are two type of malware analysis techniques which provide three different types of features: 1) Static features; 2) Dynamic features; and 3) Hybrid features. Each of three feature type is explained further.

1)Static features: Static analysis is method of malware analysis, in which sample are analyzed statically i.e. without executing the sample and only structural and physical property are analyze. Static feature are those features which are extracted by aforementioned static analysis method. Static analysis are safe and fast because sample are not executed hence the analysis platform will not be affected and so many samples can be analyzed without cleaning the analysis environment. Static feature are easy to extract without it doesn't not require complex execution and monitoring process.

2)Dynamic features: Dynamic analysis is process of executing the sample, monitoring the analysis environment and recording the changes made during the execution time. Dynamic features are those features which are extracted from the recorded changes of dynamic analysis. Dynamic analysis is time consuming and complex but it handles many of the limitations of static analysis such as it enable to extract features from packed and obfuscated malware sample which can't handle by static analysis.

3)Hybrid features: Hybrid features are combination of static and dynamic features. Integrating static and dynamic features enrich the discriminative power of feature set and improve the performance of the malware classifier. Although it's very beneficial but same time it is very costly in term of analysis time and computing.

ISBN: 978-0-9957075-1-1

III. Features for Personal Computing

Personal computing has a larger user base because of it use in various domains. Windows OS is leading with respect to number of users. Large number of users attracts the attackers which results in huge number of malware targeting only Windows user. Similarly, detection solution is also centric toward Windows malware. In this section different features are listed and explain which are used to build malware classifier.

A. Strings

Every executable or any other files have "strings" in it source which have been used as feature for building malware classifiers. All strings present in source files are extracted by static or dynamic methods and used as features following text classification approach. Static method for strings extraction has explained in [5], [6] while strings collected during "runtime trace" (by dynamic analysis) have been explained and used in [7].

B. DLL & API Call

DLL and API call are also used as features for malware detection. These two can be extracted by both static and dynamic analysis method. DLL and API are used as Boolean features, which is prepared by extracting all DLL and API call from malware and benign class and taken as features. Present and absent of DLL and API use in a sample is considered as '1' and '0' respectively [8], [9].

C. Byte-n-grams

Byte-n-grams use the frequency of "n" consecutive bytes in hexadecimal representation of a given sample as feature. Byte-n-grams are achieved by static analysis in two steps: 1) Converting sample to its hexadecimal representation, and 2) Processing and extracting byte-n-grams. Byte-n-grams based feature set is frequently used to build malware classifier [5], [10]–[14].

D. Opcode-n-grams

Opcode-n-grams use the frequency of "n" consecutive opcode in assembly representation of a given sample as feature. Opcode-n-grams are achieved by static analysis in two steps: 1) Disassemble the sample, and 2) Processing and extracting opcode-n-grams. Opcode-n-grams based features have two variants, one with consider operand along with opcode and other which doesn't take operand in consideration [12], [15]–[20].

E. PE Header Fields

PE headers fields values are also used as feature set for building malware classifier. This feature is not applicable to all file types but limited to all PE file format such DLL, exe etc. DOS_HEADER, FILE_HEADER and OPTIONAL_HEADER are three main headers from which fields value are extracted and used as feature. Various approach existed to utilize these values but all of them are carried out by static analysis method. With variation classification performance vary [8], [9], [21]–[27].

F. Network and Host Activity

Dynamic analysis provides way to monitored and extracted dynamic behaviors such as network and host activities. During the analysis time every interaction of network and host is monitored and recorded. From recorded files various kind of features are extracted and used for building machine learning based malware classifier [28]–[30].

G. Image Properties

Image properties are being used as features in image classification domain but by converting binary files to image can tap this potential for malware classification. With this motivation Nataraj et al. have converted binary file to grayscale image and then extracted GIST features from image to build malware classification system [31].

H. Hardware Features

Hardware activity is bottom of any program execution and so monitoring it gives an accurate representation of program behaviors. Wang et al. [32] have used hardware interaction based features for building malware detection system.

IV. Features for Mobile Computing

Android is leading operating system for mobile computing which spread across smart-phone to tablet. Due to a larger user base, Android is main target of security attacks and malware is one of major threat to Android. Many research works have considered this challenge and have proposed many solutions to keep Android safe from malware attacks. To the limitations of signature based detection, most of current works are focused on machine learning based Android malware detection. In this section, different features which are devised and used to build android malware detection system are listed and a short description along with appropriate work is presented.

A. Permissions and Intents

Permissions and Intents are very crucial for any Android application; it decides the app functionality and behaviors. Using permissions and intents as features resulted in high classification performance for Android malware. These two are mostly used as Boolean features but few works have considered these as numeric feature [33]–[41]. Presence and absence of permissions and intents are taken as Boolean features whereas assigned weight of each permission is taken in numeric features.

B. Strings

Similar to desktop files, Android applications also have different type of stings to accomplish various tasks. By extracting and using these strings, a feature set can be build to classify Android application into malware and benign [42].

C. System Calls

System calls are bridge between user space and kernal space, a user event results into one or more system calls. By recording system call patterns of malware and benign, a

ISBN: 978-0-9957075-1-1

Boolean feature set can be created to build Android malware classification system [34], [43]–[47].

D. Image Properties

Features extracted from image such as SIFT, GIST and HOG have demonstrated higher classification performance in computer vision domain. To utilize these features for Android malware classification, "apk to image" conversion is uses as per-processing step which convert *apk* file to image according to specified color map. From resulted image aforementioned features are extracted and used for building Android malware classifier [48].

E. Network and Host Activities

Similar to personal computing mobile's network and host activities can be recorded while a sample is in execution. Such dynamic trace provides better representation of an application behaviors hence yields better classification accuracy with machine learning models [49]–[56].

TABLE I. VARIOUS FEATURES WITH THEIR ANALYSIS TYPE

Features	Analysis Type
Personal computing	
Strings	Static & Dynamic
DLL & API call	Static
Byte-n-grams	Static
Opcode-n-grams	Static
PE headers fields	Static
Network and Host activities	Dynamic
Image properties	Static
Hardware features	Dynamic
Mobile Computing	
Permissions and Intents	Static
Strings	Static
System calls	Static & Dynamic
Image properties	Static
Network and Host activities	Dynamic

V. FEATURES FOR EMBEDDED COMPUTING

Embedded computing is getting popular due to improved hardware and advancement in software. Circuit based instruction methods are getting replaced with software alternatives which provides more functionality and flexibility (automated car, digital appliance etc.). This migration also brings threats associated with software such as malware. In recent past, few attacks on embedded system are reported but due to few users the malware problem is not getting attention. In future, with increase in user base and malware attacks this serious issue will be addressed.

VI. FEATURES FOR ICS COMPUTING

Industrial Control System (ICS) comprise computing and network infrastructure for monitoring, automating and controlling the industrial system for example nuclear power plant, water and electric distribution & controlling network etc. Affect of attacking and damaging such infrastructure will be very dangerous and not only financial loss will occur but much life will be lost. *Stuxnet* [57] is one of such malware which was written and released targeting SCADA system installed in nuclear plant. Works have started to secure ICS system but use of machine learning is very limited. Most of works are focused on patching the known weakness of computer networks and systems. Solution targeting malware is very limited but future works will benefits of machine learning based solutions for securing ICS computing environment.

VII. CONCLUSIONS AND FUTURE WORKS

In this work, we have summarized the different features used with machine learning to detect malware in various computing environment. This work provides a state-of-art status of machine learning based malware detection.

In future work, an empirical study will be conducted to validate the detection rate of various features and will try to filter out the most effective and efficient features to detect malware with respect to various computing environments.

REFERENCES

[1] WebResource, "Av-Test.org," Online, 2016. [Online]. Available: https://www.av-test.org/en/statistics/malware/. [Accessed: 03-Dec-2016].

[2] P. Košinár, J. Malcho, and R. Marko, "Av testing exposed," no. September, pp. 1–7, 2010.

[3] D. Dagon and P. Vixie, "AV Evasion Through Malicious Generative Programs," pp. 1–6.

[4] S. Alvarez and T. Zoller, "The death of AV defense in depth?-revisiting anti-virus software," 2008.

[5] M. G. M. G. Schultz, E. Eskin, F. Zadok, and S. J. S. J. Stolfo, "Data mining methods for detection of new malicious executables," Proceedings 2001 IEEE Symposium on Security and Privacy, 2001, pp. 38–49.

[6] A. Shabtai, R. Moskovitch, Y. Elovici, and C. Glezer, "Detection of malicious code by applying machine learning classifiers on static features: A state-of-the-art survey," Inf. Secur. Tech. Rep., vol. 14, no. 1, pp. 16–29, Feb. 2009.

[7] K. Rieck, T. Holz, C. Willems, D. Patrick, P. Düssel, and P. Laskov, "Learning and classification of malware behavior," in Detection of Intrusions and Malware, and Vulnerability Assessment, Springer, 2008, pp. 108–125.

[8] M. Narouei, M. Ahmadi, G. Giacinto, H. Takabi, and A. Sami, "DLLMiner: Structural mining for malware detection," Secur. Commun. Networks, vol. 8, no. 18, pp. 3311–3322, 2015.

ISBN: 978-0-9957075-1-1

[9] T. Y. Wang, C. H. Wu, and C. C. Hsieh, "Detecting unknown malicious executables using portable executable headers," NCM 2009 - 5th Int. Jt. Conf. INC, IMS, IDC, pp. 278–284, 2009.

[10] N. Nissim, R. Moskovitch, L. Rokach, and Y. Elovici, "Novel active learning methods for enhanced PC malware detection in windows OS," Expert Syst. Appl., vol. 41, no. 13, pp. 5843–5857, Oct. 2014.

[11] R. K. Shahzad, S. I. Haider, N. Lavesson, and R. K. Shazhad, "Detection of spyware by mining executable files," in Availability, Reliability, and Security, 2010. ARES'10 International Conference on, 2010, pp. 295–302.

[12] A. Shabtai, R. Moskovitch, C. Feher, S. Dolev, and Y. Elovici, "Detecting unknown malicious code by applying classification techniques on OpCode patterns," Secur. Inform., vol. 1, no. 1, pp. 1–22, 2012.

[13] R. Moskovitch, C. Feher, N. Tzachar, E. Berger, M. Gitelman, S. Dolev, and Y. Elovici, "Unknown malcode detection using OPCODE representation," in Intelligence and Security Informatics, Springer, 2008, pp. 204–215.

[14] T. Abou-Assaleh, N. Cercone, V. Keselj, and R. Sweidan, "Detection of New Malicious Code Using N-grams Signatures.," PST, pp. 193–196, 2004.

[15] R. K. Shahzad, N. Lavesson, and H. Johnson, "Accurate Adware Detection Using Opcode Sequence Extraction," 2011 Sixth Int. Conf. Availability, Reliab. Secur., pp. 189–195, Aug. 2011.

[16] R. K. Shahzad and N. Lavesson, "Veto-based malware detection," in Availability, Reliability and Security (ARES), 2012 Seventh International Conference on, 2012, pp. 47–54.

[17] P. O'Kane, S. Sezer, K. McLaughlin, and E. G. Im, "SVM Training Phase Reduction Using Dataset Feature Filtering for Malware Detection," IEEE Trans. Inf. Forensics Secur., vol. 8, no. 3, pp. 500–509, Mar. 2013.

[18] M. E. Karim, A. Walenstein, A. Lakhotia, and L. Parida, "Malware phylogeny generation using permutations of code," J. Comput. Virol., vol. 1, no. 1–2, pp. 13–23, Sep. 2005.

[19] I. Santos, F. Brezo, B. Sanz, C. Laorden, and P. G. P. G. Bringas, "Using opcode sequences in single-class learning to detect unknown malware," IET Inf. Secur., vol. 5, no. 4, pp. 220–227, 2011.

[20] A. Lakhotia, A. Walenstein, C. Miles, and A. Singh, "VILO: a rapid learning nearest-neighbor classifier for malware triage," J. Comput. Virol. Hacking Tech., vol. 9, no. 3, pp. 1–15, Mar. 2013.

[21] M. Belaoued and S. Mazouzi, "A Real-Time PE-malware Detection System Based on CHI-Square Test and PE-File Features," in IFIP Advances in Information and Communication Technology, 2015, vol. 456, pp. 416–425.

[22] B. David, E. Filiol, and K. Gallienne, "Structural analysis of binary executable headers for malware detection optimization," J. Comput. Virol. Hacking Tech., pp. 1–7, 2016.

[23] Z. Markel and M. Bilzor, "Building a Machine Learning Classifier for Malware Detection."

[24] R. Merkel, T. Hoppe, C. Kraetzer, and J. Dittmann, "Statistical detection of malicious PE-executables for fast offline analysis," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6109 LNCS, pp. 93–105, 2010.

[25] A. Walenstein, D. J. Hefner, and J. Wichers, "Header information in malware families and impact on automated classifiers," Proc. 5th IEEE Int. Conf. Malicious Unwanted Software, Malware 2010, pp. 15–22, 2010.

[26] G. Yan, N. Brown, and D. Kong, "Exploring discriminatory features for automated malware classification," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 7967 LNCS, pp. 41–61, 2013.

[27] J. H. Yang and Y. Ryu, "Toward an Efficient PE-Malware Detection Tool 1," vol. 109, pp. 14–17, 2015.

[28] U. Bayer, P. M. Comparetti, C. Hlauschek, C. Kruegel, and E. Kirda, "Scalable, Behavior-Based Malware Clustering.," in NDSS, 2009, vol. 9, pp. 8–11.

[29] C. Yavvari and A. Tokhtabayev, "Malware characterization using behavioral components," Comput. Netw. …, 2012.

[30] I. Gurrutxaga, O. Arbelaitz, J. M. Perez, J. Muguerza, J. I. Martin, and I. Perona, "Evaluation of malware clustering based on its dynamic behaviour," in Proceedings of the 7th Australasian Data Mining Conference-Volume 87, 2008, pp. 163–170.

[31] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware Images : Visualization and Automatic Classification," 2011.

[32] X. Wang, S. E. K. Chai, M. Isnardi, S. Lim, and R. Karri, "Hardware Performance Counter-Based Malware Identification and Detection with Adaptive Compressive Sensing," ACM Trans. Archit. Code Optim., vol. 13, no. 1, 2016.

[33] B. Sanz, I. Santos, C. Laorden, X. Ugarte-Pedrero, and P. G. Bringas, "On the automatic categorisation of android applications," 2012 IEEE Consum. Commun. Netw. Conf. CCNC'2012, pp. 149–153, 2012.

[34] P. P. K. Chan and W. K. Song, "Static detection of Android malware by using permissions and API calls," Proc. - Int. Conf. Mach. Learn. Cybern., vol. 1, pp. 82–87, 2015.

[35] S. Ju, H. Seo, and J. Kwak, "Research on android malware permission pattern using permission monitoring system," Multimed. Tools Appl., 2016.

[36] T. Nandhini and V. Arulmozhi, "Permission Tracking Security Model in Android Application," vol. 4, no. 2, pp. 6–12, 2015.

[37] F. Di Cerbo, A. Girardello, F. Michahelles, and S. Voronkova, "Detection of malicious applications on android OS," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 6540 LNCS, pp. 138–149, 2011.

[38] H.-Y. Chuang and S.-D. Wang, "Machine Learning Based Hybrid Behavior Models for Android Malware Analysis," 2015 IEEE Int. Conf. Softw. Qual. Reliab. Secur., pp. 201–206, 2015.

[39] S. B. Almin and M. Chatterjee, "A novel approach to detect Android malware," Procedia Comput. Sci., vol. 45, no. C, pp. 407–417, 2015.

[40] M. Magdum, "Permission based Mobile Malware Detection System using Machine Learning Techniques," vol. 14, no. 6, pp. 6170–6174, 2015.

[41] S. Verma and S. K. Muttoo, "An Android Malware Detection Framework-based on Permissions and Intents," vol. 66, no. 6, pp. 618–623, 2016.

[42] A. MartIn, H. D.Menendez, and David Camacho, "String-based Malware Detection for Android Environments," in Intelligent Distributed Computing X, 2017, vol. 678, pp. 99–108.

[43] Dimjaševic, Marko, Simone Atzeni, Ivo Ugrina, and Zvonimir Rakamaric. "Android malware detection based on system calls." University of Utah, Tech. Rep (2015).

[44] M. Dimjaševi, S. Atzeni, and Z. Rakamari, "Evaluation of Android Malware Detection Based on System Calls," 2016.

[45] D. Arp, M. Spreitzenbarth, H. Malte, H. Gascon, and K. Rieck, "Drebin: Effective and Explainable Detection of Android Malware in Your Pocket," in Symposium on Network and Distributed System Security (NDSS), 2014, pp. 23–26.

[46] S. K. Dash, G. Suarez-tangil, S. Khan, K. Tam, M. Ahmadi, J. Kinder, and L. Cavallaro, "DroidScribe : Classifying Android Malware Based on Runtime Behavior," Mob. Secur. Technol., no. October, 2016.

[47] Y. Aafer, W. Du, and H. Yin, "DroidAPIMiner: Mining API-Level Features for Robust Malware Detection in Android," Secur. Priv. Commun. Networks, vol. 127, pp. 86–103, 2013.

[48] A. Kumar, K. P. Sagar, K. S. Kuppusamy, and G. Aghila, "Machine learning based malware classification for Android applications using multimodal image representations," 2016 10th Int. Conf. Intell. Syst. Control, pp. 1–6, 2016.

[49] M. K. Alzaylaee, S. Y. Yerima, and S. Sezer, "DynaLog : An automated dynamic analysis framework for characterizing Android applications," 2017.

[50] U. Zurutuza and N.-T. Simin, "Behavior-based malware detection system for the Android platform," 2011.

[51] M. Y. Wong and D. Lie, "IntelliDroid : A Targeted Input Generator for the Dynamic Analysis of Android Malware," no. February, pp. 21–24, 2016.

[52] A. Ali-gombe, I. Ahmed, and G. G. R. Iii, "AspectDroid : Android App Analysis System," pp. 145–147.

[53] H. Wang, Y. Guo, Z. Tang, G. Bai, and X. Chen, "Reevaluating Android Permission Gaps with Static and Dynamic Analysis," 2015.

[54] M. Spreitzenbarth, F. C. Freiling, F. Echtler, T. Schreck, and J. Hoffmann, "Mobile-Sandbox: Having a Deeper Look into Android Applications," ACM Symp. Appl. Comput., pp. 1808–1815, 2013.

[55] E. B. Siegfried Rasthofer, Steven Arzt, Marc Miltenberger, "Harvesting Runtime Data in Android Applications for Identifying Malware and Enhancing Code Analysis," Ndss, 2016.

[56] M. Zheng, M. Sun, and J. C. S. Lui, "DroidTrace : A Ptrace Based Android Dynamic Analysis System with Forward Execution Capability," pp. 1–6.

[57] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," Secur. Privacy, IEEE, vol. 9, no. 3, pp. 49–51, 2011.

AUTHOR PROFILE

Ajit Kumar is a Ph.D. Research Scholar in the Department of Computer Science at Pondicherry Central University, Puducherry, India. He has completed his Master's degree in Computer Science in the year 2011 and Bachelor's degree in Computer Application in year 2009. His research interest includes Cyber Security, Malware Classification and Machine Learning. He has published 6 papers in International Conference realted to Malware and Machine Learning.

K.S.Kuppusamy is an Assistant Professor of Computer Science at Pondicherry Central University, India. He has received his Ph.D. in Computer Science and Engineering in the year 2013 and his master's degree in Computer Science and Information Technology in the year 2005. He has got a total of 11 years of teaching experience. His research interest includes Accessible Computing, Security and accessibility. He has published more than 25 papers in various international journals and coferences. He is the recipient of Best Teacher award during the years 2010, 2011 2013, 2015 and 2016.

G. Aghila is Professor at Department of Computer Science and Engineering, National Institute of Technology Puducherry, Karaikkal. She has got a total of 26 years of teaching experience. She has received her M.E (Computer Science and Engineering) and Ph.D. from Anna University, Chennai, India. She has published nearly 70 research papers in web crawlers and ontology based information retrieval. She has successfully guided 7 Ph.D. scholars. She was in receipt of Schrneiger award. She is an expert in ontology development. Her area of interest includes Intelligent Information Management, Artificial intelligence, Text mining and Semantic web technologies.

ISBN: 978-0-9957075-1-1

International Journal of Engineering and Applied Computer Science (IJEACS)

Comparison of Health Care System Architectures

Mahboobeh Abdoos
Department of Electrical and Computer Engineering
Qom Islamic Azad University
Qom, Iran.

***Abstract*—Body area sensor network is an important technology which is suitable for monitoring the patient's health and real time diagnosing the diseases. The body area network includes the sensors which can be spread over the body or the wearable cloth and a coordinator node which can be a mobile or a tablet or a PDA, which receives the signal of a person's sensors. In the new architecture the coordinator node sends the information to the central data server via internet or GPRS or MANET. The central data server is responsible for saving and analyzing and representing the received data in the text and graphical mode and sending SMS to the patient's family or the nearest ambulance or physician, or the operator can call them. The received information is analyzed by the data mining tools. The necessary information will be sent to the physician's computer. Every patient has a GPS, and it is supposed that the encryption is used for transferring information. In this paper the new architecture is compared with the traditional one which includes the base station and relay nodes. It is shown that the new architecture has less delay than the traditional one.**

Keywords — Architecture; Health care; Sensor; MANET

I. Introduction

The body sensor network can help people by preparing health care services, like monitoring and communication via SMS or GPRS. The health monitoring system uses the wearing cloth which has sensors or the body implanted sensors. The health care system helps the patients and their families by monitoring their physiological signal without interrupting the patient's normal life and increasing the quality of patient's life. The health care system does not limit a patient to stay in the bed and in the current architecture, the patient's physiological signal is received by the patient's sensors and it is transferred to the coordinator node which is a mobile or tablet, then it is transferred to the far base station and then to a computer to save and analyze them. In the close environments, the signal length weakens. Increasing of Obstacles between nodes causes the increasing of the packets loss ratio. So it is needed to increase the relay nodes, in the closed environment, to cover the entire of environment. This architecture is dependent to the infrastructure and relay nodes and base stations and suffers the infrastructure cost.

The proposed architecture eliminates the infrastructure and the sensor nodes transfer their information to the coordinator node and the coordinator sends the information to the central data server via internet or GPRS. The proposed architecture has less cost and less delay than the old architecture. In this paper, section 2, surveys the ad hoc network and some of the routing protocols of it. Section 3 reviews the body area sensor network and health care system architectures. Section 4 represents the simulation result of two health care system architectures comparison. There is shown that the new architecture which uses the internet or GPRS or MANET to transfer the signal information of coordinator node (the coordinator is gathering the signal information of sensors) to the central data server, has less delay in comparison with the traditional architecture which uses the base station and relay nodes to transfer the information of signal to the central data server [1], [2], [3], [4].

II. Ad Hoc Network

An ad hoc network consists of some wireless mobile nodes which route the packets without any infrastructure. The ad hoc network is divided to static and dynamic ad hoc networks. In a static ad hoc network, the location of a node does not change. In the dynamic ad hoc networks the nodes are moving like the mobile and vehicle ad hoc networks. The topology of the mobile ad hoc network is changing. There are two kinds of routing, the first one is the topology based routing and the second one is the location based routing. The topology based routing uses the information of links of the network to transfer the packets. It is divided to the table-driven and demand based routing protocols. The table-driven routing protocols consist of the distance-vector protocols and the link-state protocols [5], [6], [7], [8].

III. Routing Protocols In Ad Hoc Networks

A. Ad hoc On-Demand Distance Vector Routing protocol (AODV)

AODV uses the combination of demand based routing (DSR) and hop by hop routing (DSDV). It uses the sequential number in the table of node. This number is

ISBN: 978-0-9957075-1-1

produced by the destination node. This number is not in the route request packet and in the route reply packet and it is sent to the requested nodes. This number is so important because it avoids the loop and the other node uses this number to update its routing information. AODV is consistent with the routing tables. The route request packets and the route reply packets and errors are defined like DSR. The privilege of this approach is the proper throughput, but it can not find and support and update the long paths.

B. Destination Sequenced Distance Vector Routing protocol (DSDV)

DSDV is the changed Bellman Ford algorithm. Every node has the entries for the destination node, which consists of the next hop and the number of hops to the destination node. Every node propagates its routing table to its neighbors to update them. The privilege of DSR is saving the fix paths to all other nodes of network by a node, but it causes the wasting of band width and saving the useless paths that may never are used.

C. Dynamic Source Routing Protocol (DSR)

DSR is a routing protocol which can manage the ad hoc network without needing to routing tables and updating them. To save the band width, it is done, when just it is needed. In DSR, the source node defines all of the routes from source node to the destination node and saves the path of intermediate nodes. It is a link state algorithm. Every node saves the best route to the destination node. If any change is occurred in the network, all nodes of the network are informed via broadcasting the changes. DSR does not need to update periodically. The control overhead is little, which causes saving the band width.

IV. New Architecture

This architecture includes four parts

A. Sensor Nodes

The sensor nodes monitor the main body parameters, which show the patient's health or sickness, like the body temperature, heartbeat, blood pressure, breathing ratio, blood oxygen. For example the sensors for monitoring heart beat are EGG, the sensors for sensing signal of the brain are EEG and the sensors for sensing signal of the muscles are EMG.

B. Coordinator Node

The coordinator node is a wireless node in the BANET, which is responsible for receiving signal of sensors and sending the information to the central data server. This node can be identified by a unique patient's ID. This node can be a mobile or tablet that uses the internet or GPRS or MANET to send the information to the central data server. (See figure 1) [9], [10], [11], [12]

C. Central Data Server

This server saves the received information to process later. The data mining is a useful tool for analyzing the huge amount of information. The graphical user interface in the central data server is responsible for saving and analyzing and representing the received data in the text and graphical mode and sending SMS to the patient's family or the nearest ambulance or physician or the operator can call them. The figure 2 shows the new architecture that the patients in the different locations can use this health care system.

The privilege of this architecture is that it removes the infrastructure cost of traditional architecture (removes the base station and relay nodes). The communication between the coordinator node and the central data server is via internet or GPRS or MANET and the routing protocols in MANET like DSDV, DSR, and AODV are used.

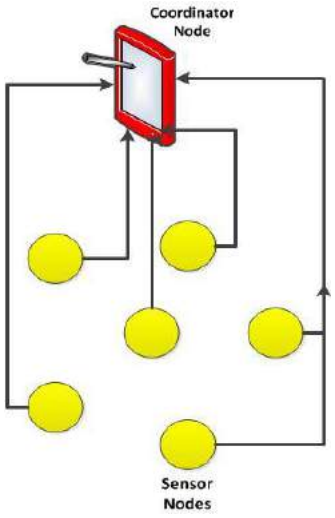


Figure 1. Sensor nodes in the body or wearable cloth and coordinator node which can be a mobile or a tablet.

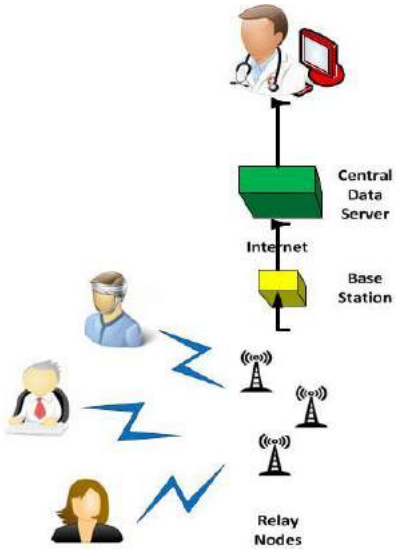


Figure 2. Traditional health care system architecture which uses the relay nodes to transfer information to the central data server.

ISBN: 978-0-9957075-1-1

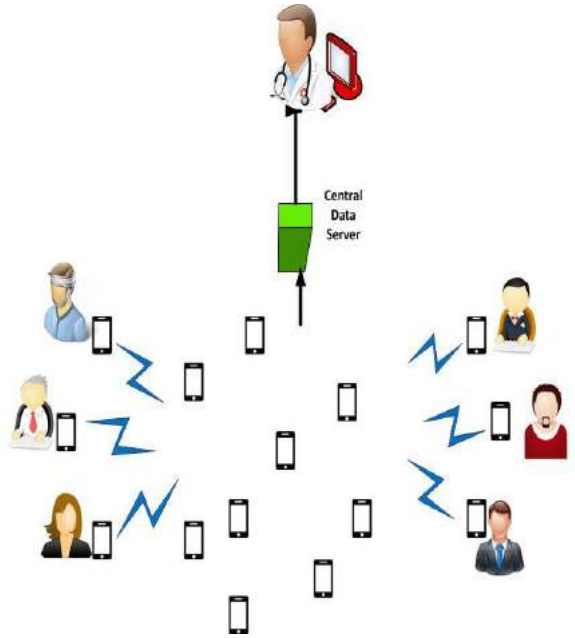


Figure 3. New health care system architecture which does not use the relay nodes to transfer information to the central data server

V. The Simulation Environment

The simulation is done in NS2. The simulation scope is 1500×1500 meters in 200 seconds. In this simulation, it is used the IEEE 802.11 protocol in the MAC layer and the band width is 2 Mbps. The data packets size is 512 byte and the CBR traffic is used. The packets sending rate is 4 packets in a second. In simulation running, the random way point model is used for moving the nodes. The nodes maximum radio range is 200 meters. The nodes number in this simulation is 100 and the nodes speed are 10 m/s and 1 m/s for the first and second scenarios. As it is shown, in the figure 4, when the speed of nodes is increasing the probability of fast packets transferring to the neighbor nodes and the delay is increasing. By comparing the architectures, it is shown that the new architecture has less delay than the traditional architecture.

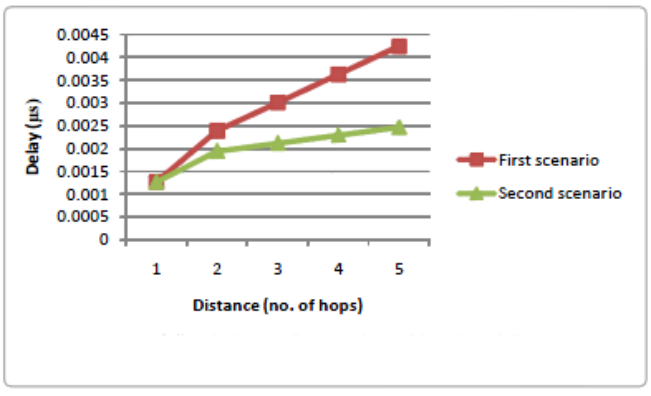


Figure 4. Scenarios delay comparison of new architecture.

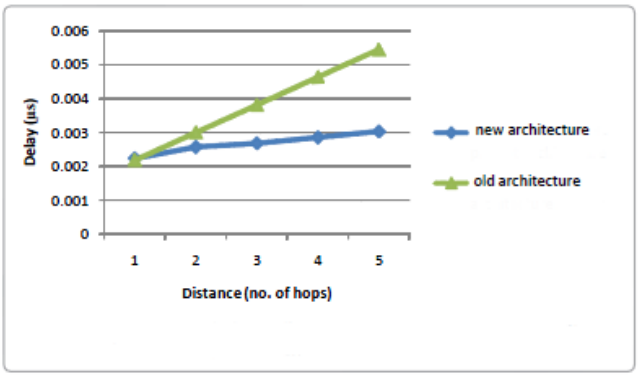


Figure 5. Delay comparison of two architectures

VI. Conclusion

Wireless sensor networks (WSNs) are finding applications in many areas, such as medical monitoring, emergency response, security, industrial automation, environment and agriculture, seismic detection, infrastructure protection and optimization, automotive and aeronautic applications, building automation, and military applications. The body area sensor network is an important technology that is suitable for monitoring the patient health and real time diagnosing the diseases. The body area network includes the sensors which can be spread over the body or the wearable cloth and a coordinator node that can be a mobile or a tablet or a PDA that receives the person sensors signal. In the new architecture the coordinator node sends the information to the central data server via internet or GPRS or MANET. The central data server is responsible for saving and analyzing and representing the received data in the text and graphical mode and sending SMS to the patient's family or the nearest ambulance or physician, or the operator can call them. The received information is analyzed by the data mining tools. The necessary information will be sent to the physician's computer. Every patient has a GPS, and it is supposed that the encryption is used for transferring information. In this paper the new architecture is compared with the traditional one which includes the base station and the relay nodes. It is shown that the new architecture has less delay than the traditional one.

References

[1] Stanford V, "Using pervasive computing to deliver elder care", IEEE Pervasive Computing 10-13, 2002.

[2] Mcfadden T, Indulska J ,"Context-aware environments for independent living", In Proceedings of the 3rd National Conference of Emerging Researchers in Ageing, Brisbane, Australia, 2004.

[3] Anliker U, Ward JA, Lukowicz P, Troster G, Dolveck F, et al. "AMON: a wearable multi-parameter medical monitoring and alert system", IEEE Trans Inf Technol Biomed 8: 415–427, 2004.

[4] Cho G, Yoo1 SK, "Wearable ECG Monitoring System Using Conductive Fabrics and Active Electrodes", Proceedings of the 13th International Conference on Human-Computer Interaction, Berlin, Heidelberg, 2009.

ISBN: 978-0-9957075-1-1

[5] Darwish A, Hassanien AE ,"Wearable and Implantable Wireless Sensor Network Solutions for Healthcare Monitoring", Sensors 12: 12375-12376, 2012.

[6] Shnayder V, Chen B, Lorincz K, FulfordJones TRF, Welsh M ,"Sensor Networks for Medical Care", Proceedings of the 3rd international conference on Embedded networked sensor systems, New York, USA, 2005.

[7] Alemdar H, Ersoy C ,"Wireless sensor networks for healthcare: A survey", Computer Networks 54: 2688-2710, 2010.

[8] Varshney U ,"Using wireless networks for enhanced monitoring of patients", International Journal of HealthCare Technology and Management 6: 489-499, 2005.

[9] Varshney U ," Pervasive healthcare", IEEE Computer 12: 138-140, 2003.

[10] Lai CC, Lee RG, Hsiao CC, Liu HS, Chen CC ,"A H-QoS-demand personalized home physiological monitoring system over a wireless multi-hop relay network for mobile home healthcare applications". Journal of Network and Computer Applications 32: 1229-1241, 2009.

[11] Chung WY, Walia G, Lee YD, Myllyla R ,"Design Issues and implementation of Query-Driven Healthcare System Using Wireless Sensor Ad-hoc Network", IFMBE Proceedings 13: 99-104, 2007.

[12] Hande A, Polk T, Walker W, Bhatia D, "Self-Powered Wireless Sensor Networks for Remote Patient Monitoring in Hospitals", Sensors 6: 1102-1117, 2006.

AUTHOR PROFILE

Mahboobeh Abdoos received the B.S and M.S degrees in computer engineering from Azad University, Ghazvin, Iran, in 2002 and 2007 respectively. She is now the Ph.D. research student of Islamic Azad university, Qom, Iran. She has taught at Islamic Azad and Payam Nour Universities from 2005 til now. She has been the referee of some conferences. Her current research interest includes position based routing protocols in mobile ad hoc networks, QOS and security based routing protocols in mobile ad hoc networks, cloud computing and data base.

EMPIRICAL RESEARCH PRESS

International Journal of Engineering and Applied Computer Science (IJEACS)

Understanding the Cloud Computing: A Review

Kamalinder Kaur
Assistant Professor
Dept. of Computer Science & Engineering
Chandigarh Engineering College, Landran
Punjab, India.

Nupur
Assistant Professor
Dept. of Computer Science & Engineering
Chandigarh Engineering College, Landran
Punjab, India.

***Abstract* - Now a days the work is being done by hiring the space and resources from the cloud providers in order to do work effectively and less costly. This paper describes the cloud, its challenges, evolution, attacks along with the approaches required to handle data on cloud. The practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer. The need of this review paper is to provide the awareness of the current emerging technology which saves the cost of users.**

Keywords— cloud; SAAS; providers; SOA; resources.

I. INTRODUCTION

The term "cloud" deals with the group of system which exchanges the data from one system to another by providing resources over the internet. The resources existing in cloud can be used enormously by user whenever they need it. In cloud computing, physical infrastructure is usually desired to third party provider for service of internet instead of setting up their own resources by the customers. Customers use the resources as a service and pay only for the used quantity of resources. Cloud providers use self-service abilities and virtualization technologies for allocating resources via network infrastructure. In cloud environment, many kinds of virtual machines are put on the same physical server for communications. In cloud, customers are paid for what they used and are not paid for storage or infrastructure considered as local resources.

II. LITERATURE SURVEY

"Evolution of Cloud Computing, its Approaches and Comparison with Grid Computing" Rajleen Kaur, Amanpreet Kaur. In this review paper they have discussed about the cloud computing characteristics and its evolution. As in this the cloud computing and its approaches are studied, the cloud computing will increase in today's scenario. So, the cloud computing have a great impact on society. Review paper on Cloud Computing, Seema Sharma, Jyoti Godara. In this paper they discuss about various deployment model and cloud services of cloud computing. Cloud computing is recent buzzword in IT world. The Leader in the companies, such as Microsoft, Amazon, IBM, and Google provided their initiative in promoting cloud computing. But still there are some issues in the cloud computing like testing issues, security issues, and privacy issues.

Research Agenda in Cloud Technologies. Ilango Sriram, Ali Khajeh-Hosseini. This paper has presented the work published by the academic community advancing the technology of cloud computing. Much of the work has focused on creating standards and allowing interoperability, and describes ways of designing and building clouds.

III. KINDS OF CLOUD COMPUTING

A. Public Cloud

Standard models providers to create many resources such as applications and storage, which are available to the public. Public cloud services are either free or not. Public clouds which are running applications outwardly by large service providers, provides some benefits over private clouds.

B. Private Cloud

It identifies to internal services of a business that is not accessible for regular people. Private cloud is a promoting term for a structural design that provides hosted services abaft the firewall to the people.

C. Hybrid Cloud

A domain that company offers and controls some resources internally and has some others for public consumption. It is a group of private and public clouds. Cloud provider, has a service that uses private cloud and is only authorized by certified staff and is protected by firewalls from outside accessing. Also using a public cloud environment in which external users can access to it. In cloud computing, the capacity is lifted so that workload can be reduced. For successively working applications, the local computers do not to take the heavy load. Actually this burden is handled by group of computer which forms the cloud. On Customer side the demand of hardware and software reduces. So, the thing

ISBN: 978-0-9957075-1-1

which is needed to be considered for software on computer is the web browsers, like Google chrome, opera, Mozilla Firefox etc. Structures of cloud computing consists of on-demand self-services, broad network area, measure services, rapid elasticity, reduced pooling, multi-tenacity and sharing of infrastructure. The Cloud computing requires Software, hardware, application stage, arrangement and storage with an internet connection.

IV. Methodologies Of Cloud Computing

A. SaaS-Software as a Service

B. PaaS-Platform as a Service

C. IaaS-Infrastructure as a Service

Public customers have voluntarily included cloud computing facility like Google-email, Facebook, YouTube, yahoo, Hotmail, Twitter etc. It provides decline in management duties and the main focus is on innovation and production. It is widely used in business which meets the requisite of varying environment. Many firms provide services from the cloud like Yahoo, Google, Microsoft, SalesForce.com, Amazon etc.

V. Growth Of Cloud Computing

There is an involvement of research in defining the cloud computing and Youssef et al. were first among the persons to give knowledge of cloud computing concept along with its modules. According to the researchers it is an arrangement of concepts in research fields like SOA, grid computing, virtualization, distributed computing. Conferring to Youseff, "cloud computing is stated as a new computing archetypal that allows users to temporarily employ computing infrastructure over the network, restored as a service by the cloud-provider at feasibly one or more levels of abstraction" (Youseff et al. 2008). Conferring to Armbrust et al. "Cloud Computing refers to both the applications carried as services over the Internet and the hardware and systems software in the data centers that offer those services. The services themselves have long been denoted as Software as a Service (SaaS). It's the data center hardware and software is what we say a Cloud. When a Cloud is made available in a pay-as-you-go mode to the general public, is called as a Public Cloud, the facility being sold is Utility Computing. We use the term Private Cloud to refer to internal datacenters of a business or other organization, not obtainable to the general public. Therefore, Cloud Computing is the totality of SaaS and Utility Computing, but does not include Private Clouds"

VI. Features

As per the meaning of the cloud computing there are generally five essential characteristics of the clouds. It is understandable that missing any of these characteristics is not considerable in it.

A. On Demand Capabilities

In business, cloud computing provider will secure the cloud computing services, it considered as a software vendor. The user can access services and can change them through online control panel without interacting with the server or it can be done straight with the provider.

B. Wide Network Access

Now a day, all digital devices like tablet, mobiles, laptops, etc. can access broad networks whenever they come in connect with a simple network access point. In business, this feature is very valuable because employees can stay connected with contracts, proposals, projects and customers during office hours even off times.

C. Resources Pooling

In the cloud computing environment, the employee can share the data or services simultaneously from any location at any time within business management software hosted at the cloud.

D. Rapid Elasticity

Cloud computing also offers the flexibility and scalability up to that extent that you can add or remove the users and services as per your need.

E. Measured Service

The cloud computing is so affordable that you can access services and for what type of service you used you have to pay for it. It can be examined from both the sides including source's side and user's side and hence improves transparency.

VII. Characteristics

Infrastructure Sharing is where the user shares the infrastructure on a virtualized concept, allowing the sharing of physical utilities, storage, networking proficiencies.

A. Self-motivated Provisioning

It allows services on the basis of requirement of present demands. It can be done automatically using automation, allowing the enlargement and reduction of service proficiency. The self- motivating scaling is required to be done by maintaining the great levels of reliability and security.

B. Access of Network

It requires to be called up across the internet from a variety of devices such as desktops, laptops, palmtops, tablets, mobile devices, using standards-based APIs (such as HTTP). Setting out of services in the cloud consists of everything from using business applications to the latest applications on smartphones.

C. Metering Management

It is helpful in order to generate the reports and create the bills for the reading, the users has used the resources during that period of assignment. It gives optimized service. They have really used during the billing period.

ISBN: 978-0-9957075-1-1

In essence, cloud computing allows the desirable work doing from any place any time any location for which the user has to pay about the actual usage.

VIII. SERVICE MODELS

The following are the models which are required after the cloud is being deployed.

A. Saas - Software As A Service

Consumer is the payee on the amount he accesses and uses a request and facility provided by the cloud. For example Google apps, is a source in which information is shared and interacted between the customer and provider. Also the contribution of Microsoft is remarkable in this area, and as part of the cloud computing option for Microsoft® Office 2010, through its cloud-based Online Services, its Applications are obtainable to those customers having Office volume licensing and Office Web Applications subscriptions.

B. Paas - Platform As A Service

To install software and applications in the cloud, customers purchase access to the platforms. Consumers do not manage operating systems and network accesses, and there might be restrictions as to which applications can be deployed.

C. Iaas - Infrastructure As A Service

Customers do not control the cloud infrastructure, but can perform controlling and monitoring of the systems in terms of operating systems, application software, storage, and network connections.

Another subset known as **CaaS-Communication as a Service.** It is a meant for hosted IP telephony services. In this context, CaaS may be viewed as a subset of **SaaS- Software as a Service (SaaS) -** End user application is delivered as a service. There is an abstraction of platform and infrastructure and less effort is required to deploy CaaS.

Platform as a Service (PaaS) - Custom applications and services can be deployed through this application platform inexpensive application can be built and deployed easily, though managed services are needed.

Infrastructure as a Service (IaaS) - The cost and necessity for dedicated systems is minimized as physical infrastructure is abstracted to deliver computing, storage, and networking as a service.

IX. BENEFITS

A. Cost Savings

To get a hike in computing capabilities, companies can lessen their capital expenditures and use operational ones. It requires less in-house information technology resources to provide system support for the growth of company.

B. Scalability/Flexibility

Companies can start their setup from scale to large .In order to fulfill the consumer demand, companies can use extra resources if required because of scalability feature provided by cloud computing.

C. *Reliability*

It supports disaster management facility and provides consistency in business.

D. Maintenance

In order to avoid application installations onto PCs, system maintenance is performed by cloud service providers and through Application Program Interface, the services are accessed which helps in reducing maintenance cost.

E. Mobile Accessible

Productivity of mobile workers has been increased due to availability of resources from any corner of the world.

X. CLOUD COMPUTING ISSUES

A. Technical Issues

Cloud computing needs high technology and strong internet connection. Through this technology we can access the data and information anywhere and anytime. As we know technology is always prone to some technical issues. Therefore high maintenance and good internet connection is required for using server at all the time.

B. Security In The Cloud

Before using this technology, all company's important information is shared with a group of service providers other than those directly involved in providing the services. As a result company comes under a great risk. A consistent service provider plays a vital role in securing the information.

C. Prone To Attack

The information stored in cloud is accessible to external hackers and threats. As the data is not completely secure on the internet. There are always some chances of hacking of sensitive data.

XI. CHALLENGES

The prominent challenges associated with cloud computing are described below. Although some of the following may cause a decline when adding more services in the cloud, but if handled with care and observation in planning stages, most can become an opportunity.

A. Security and Privacy

The major issues related to cloud computing that are generally due to the slowdown of deployment of cloud services includes storing and securing data, and monitoring the use of the cloud by the service providers. These can be resolved, e.g. the information is stored inside the organization, but permitting it to be accessed in the cloud. For this to occur, a Hybrid cloud could be used to support strong security mechanisms between organization and the cloud.

ISBN: 978-0-9957075-1-1

B. Lack of Standards

It is unlikely that most clouds will be interoperable as no standards are associated with the standard interfaces of clouds. In order to resolve this problem, the Open Cloud Consortium is working on cloud computing standards and practices and the Open Grid Forum is developing an Open Cloud Computing Interface. The conclusions of these groups can be concerned, but it is not sure whether they will address deployment of the services. However, the services are leveraged if the modern standards are kept up to date.

C. Continuous Evolution

Customer needs are arising day by day, therefore the necessities for interfaces, networking, and storage of information. This shows that a public cloud, constantly emerging.

D. Concerns regarding Compliance

Two of many compliance concerns affecting cloud computing includes the Sarbanes-Oxley Act (SOX) in the US and Data Protection directives in the EU. These deal with the type of data and application for which the cloud is needed. For maintaining the security and secrecy, Hybrid cloud deployment is needed which comprises of one cloud storing the data internal to the organization.

XII. ATTACKS IN CLOUD COMPUTING

As the cloud can give service to legitimate users it can also provide service to users that have spiteful purpose. For achieving the objective like a DDOS attacks against cloud itself or organizing another user in the cloud, a hacker can use a cloud to host a spiteful application. Assume that an attacker, who knows that his victim is accessing specific cloud provider, can draft an attack in opposition to his offended. This states that both attacker and victim are in same network but are using virtual machines instead of physical network.

A. DDOS Attack

Distributed Denial of Service (DDOS) attacks usually focus on high number of IP packages at particular entry point. In cloud computing the resources are shared by a number of client machines. DDOS attacks may have the possibility of having much greater impact than single rented architectures. If cloud doesn't have abundant resources to provide services to its consumers then this may cause unwanted DDOS attacks, the solution for this is increase in number of such vital resources. But problem is when bot-net is used by malicious user deliberately for a DDOS attacks. Most network countermeasures cannot distinguish good traffic from bad traffic and cannot stop the cascade of traffic and therefore cannot protect against DDOS attacks. When attacks are identified and have pre-defined signatures then Intrusion Prevention Systems (IPS) becomes active but if there is legitimate content with bad intentions, IPS becomes ineffective. Unfortunately, because attacker can easily by pass firewalls, firewalls are fragile and ineffective against DDOS attacks like IPS solutions since these are designed to allow justifiable traffic and attacks produce abundant traffic from so many different hosts for cloud, its Internet connection, is unable to hold the traffic. It is prominent to say that DDOS protection works on Network Virtualization layer rather than Server Virtualization.

B. Cloud Against DDOS Attack

DDOS attacks when launched from a botnet with large numbers of automaton machines become one of the dominant threats in the world. DDOS attack sends a substantial overflow of packet to a Web server from various hosts. In this condition, the cloud may be part of the solution. It's interesting to consider that websites having limited server possessions and facing DDOS attacks are getting the benefits of using cloud that provides more resource to tolerate such attacks.

XIII. DEPLOYMENT MODELS

Depending upon the needs, deploying of cloud computing can vary. The following are the four deployment models that include the properties which support the needs of the services and users of the clouds in specific ways

A. Private Cloud

The private cloud is meant for the internal working of a particular organization. The operation may be in-house or with a third party on the ground.

B. Community Cloud

The cloud resources are public among a numerous organizations having same interests and requests. Due to this, the capital expenditure costs for its establishment are reduced as the cost is shared among different organizations.

C. Public Cloud

The cloud service provider assigns cloud resources to the public on a commercial basis. This requires very little financial outlay, compared to the capital expenditure requirements usually associated with other deployment options, by a consumer to develop and deploy a service in the cloud.

D. Hybrid Cloud

This cloud infrastructure contains any number of clouds of any type, but the interfaces are required to allow data and/or applications to be transferred from one cloud to another. This can be a private and public arrangement of clouds that support the requirement to hold some data in an organization.

REFERENCES

[1] S. Arnold (2009, Jul.). "Cloud computing and the issue of privacy." KM World, pp14-22. Available: www.kmworld.com [Aug. 19, 2009].

[2] A Platform Computing Whitepaper. "Enterprise Cloud Computing: Transforming IT." Platform Computing, pp6, 2010.

[3] Global Netoptex Incorporated. "Demystifying the cloud. Important opportunities, crucial choices." pp4- 14. Available: http://www.gni.com [Dec. 13, 2009].

[4] Peter Mell, Timothy Grance , "The NIST Definition of Cloud Computing", Recommendations of the National Institute of

ISBN: 978-0-9957075-1-1

Standards and Technology, September 2011, Special Publication 800-145

[5] Arora Pankaj, Wadhawan C.Rubal, Er.Ahuja P.Satinder, 2012, "Cloud Computing Security Issue in Infrastructure as a Service", International Journal of Advance Research in Computer Science and Software Engineering.

[6] Habib, S. M., Hauke, S., & Ries, S. (2012) "Trust as a facilitator in cloud computing: a survey", Journal of Cloud Computing, 01-18.

[7] Parsi, K., & M.Laharika. (2013) "A Comparative Study of Different Deployment Models in a Cloud", International Journal of Advanced Research in Computer Science and Software Engineering , 3 (5), 512-515.

[8] http://www.ibm.com/developerworks/cloud/library/clcloudservieliaas/.

[9] http://www.statechmagazine.com/article/2014/03/5-important-benefits-infrastructure-service.

[10] Kuyoro, S., Ibikunle, F., & Awodele, O. (2011)"Cloud Computing Security Issue and Challenges", International Journal of Computer Networks, 3 (5), 247-255.

[11] Cloud Computing Bible, 2011, Wiley Publishing, Inc., Indianapolis, Indiana, pg. 8-9.

AUTHORS PROFILE

Kamalinder Kaur is working currently as Assistant Professor in Chandigarh Group of Colleges, Landran , Punjab, India. She has five years of teaching experience, her research interest includes Networking with specialization in Mobile Ad-hoc Network (MANET).

Nupur is working currently as Assistant Professor in Chandigarh Group of Colleges, Landran , Punjab, India. She has two years of teaching experience, her research interest includes Networking and Database Security.

International Journal of Engineering and Applied Computer Science (IJEACS)

Resource Availability Prediction in the Grid: Taxonomy and Review of State of the Art

Farrukh Nadeem
King Abdulaziz University
Jeddah, Saudi Arabia.

Mahreen Nasir
University of Hail
Hail, Saudi Arabia.

***Abstract* - Use of Grid Computing for carrying out cooperative work from distributed resources has been into practice for the past several years. Efficient execution of various tasks on the grid comes with various challenges. One of them is to ensure that a particular resource remains available during the execution of task. The dynamic nature of resources makes it even more challenging to predict resource availability for complete task duration. This paper is an attempt to address this issue by providing a comprehensive review of the existing methods along with a stated taxonomy of the approaches used.**

Keywords- resource availability; grid computing; data mining; machine learning; survey

I. INTRODUCTION

In a distributed environment, resource availability prediction is necessary for grid schedulers to have smooth execution of tasks. A resource refers to an asset which is required for task execution e.g. a computing machine, CPU cycle and memory etc. It can also be a group of machines (cluster) working together to achieve a common goal. Before starting task execution, a resource should be ready and free for use. An allocated resource cannot be re-assigned to a new task. To guarantee the task execution without interruption, required resources must be made available for the complete duration needed for task completion. Resources are owned by organizations which have pre-defined policies about resource allocation which can be either on demand (non-dedicated) or made available all the time (dedicated) to the grid. It is vital to determine the resource availability for a specific time period in the near future so that it can be assigned in response to a resource request. Predicting resource availability in grid is a challenging issue mainly because the resources are non-dedicated, dynamic and may not be fully available during the task execution. Also, as the system is distributed, it is hard to record idle or busy resources. Additionally, the node(s) in the network are transient as they join or leave the network without any notice which may result in absence of the node at any particular time. This requires prior investigation of available resources for their fair and uniform allocation to various tasks.

Many techniques had been suggested to address this problem. This survey is an effort to discuss the various proposed methods for predicting resource's availability in a grid.

The survey is aimed to:

1. Provide an extensive overview of the existing resource availability prediction techniques in grids
2. Provide taxonomy for the classification of available techniques
3. Analyze the available techniques by discussing their pros and cons

II. CLASSIFICATION OF RESOURCE AVAILABILITY PREDICTION TECHNIQUES

A. Probability Theory

Prediction approaches under this category estimate probability of resource availability in the future based on its past availability patterns. The main goal is to determine that how much time a resource will spent on each particular state (multistate model).The authors in [4] present a parametric model fitting technique (weibull method) along with two non-parametric techniques(Resample method and Binomial method) to predict machine availability duration. The authors focus on the estimation of a specific quantile for the availability distribution along with a confidence level which is related with each estimate. The goal is to facilitate the schedulers to make dynamic decisions by supporting live availability predictions. The authors verified usefulness of their proposed technique by performing lower bound quantile estimation on a synthetic fixed weibull distribution. Later in the next step, availability traces for individual machines were taken and two sets of training and test data were created respectively. Training data facilitated to find the lower bound of quantile and test data was used for accuracy verification of the estimate. The study in [17] showed the use of probability theory. Various aspects were discussed for resource availability including:

ISBN: 978-0-9957075-1-1

1. Resource online serving time: it is measured by the time when the resource is offline due to some failure, reboot or service with drawl.

2. Resource serving time: Measured by time taken by local task execution in a specific period.

3. Resource availability during task execution: Resource accessibility can be measured by using the length of waiting queue.

Resource may become unreachable because of approaching the maximum limit of number of task acceptance. Prediction of a resource is done by using probability theory and resource evaluation is performed by suggesting four availability metrics. The metrics used for availability were: offline time of a resource, task execution time (local), waiting time and waiting queue length. The authors mention that the resource offline time is almost stable and it is considered as a constant and shows an exponential distribution. The second metric which is local execution time is shown to be a random variable and it shows the pattern of poison distribution. It is assumed that if a resource has a higher idle rate then it can provide service for longer periods.

1) Multistate Model

A multistate availability model is proposed in [13] to determine resource unavailability. Semi Markov process models (SMP) were used. The proposed framework is applied using an ishare production system. The method achieved an accuracy level of 86%. The authors used response time as a performance metric. Additionally, various resource unavailability types were discussed including:

1. Unavailability Due to Excessive Resource Contention (UEC): An unavailability, e.g. due to host and guest process running parallel on same machine. Guest process may lower down host process's execution. To resolve this, there might be a need to decrease the guest process's priority. As a result guest process may be stopped or had to be shifted to another machine which may cause a failure.

2. Unavailability Due to Resource Revocation (URR): This can be caused due to sudden hardware failure or machine unavailability without any notice. Authors' main contribution is to suggest a method to predict when a resource becomes unavailable. The multistate model combines the above mentioned classes of resource unavailability. In the suggested multistate model, availability and unavailability are modeled by using some observations. These observations are recorded for determining the two resource unavailability states (as mentioned above). The URR state can be caused when there is a failure in accepting service submission. The UEC state is reached in case a host process slows down. Parameters like CPU cycles and memory use are required to monitor the slow down by setting a threshold. The priority of guest process is reduced in such a scenario. The main idea is to fix a threshold which measures the slowdown of host process. A group of host processes having different resource usage were executed together as part of experiment. A single guest process is permitted to execute at a particular time. The priority of guest process is decreased if it affects host process. If the slowdown still remains, then the guest process is held and is resumed after the congestion is over, else it is terminated. The slowdown of host process is measured by the reduction in the rate of CPU usage going above a threshold which is greater than 5%. To measure the CPU load (Lh), two thresholds Th1 and Th2 were used. The value of Th1 is set as 20% and Th2 as 60%.

The suggested method was implemented within the ishare framework which consist of host (provider) and client(consumer) nodes. On submission of a job request from client, a prediction function is called. The ishare gateway component is responsible to communicate with clients. Resource Monitor component monitors the usage of host process's CPU and memory. On receiving a job submission request, the job scheduler on the client inquires the gateway about the availability of machine during the specified future time window. According to the response, if a machine is selected, guest process execution is initiated and the Resource Manager is informed about the id of new process. Any state transition during the execution of the process is signaled to the gateway in order to take necessary actions regarding process killing or migration to another machine. A multistate grid resource availability characterization is presented in [13]. The authors propose a multistate model to determine the future availability of a resource. The model has five states of resource availability which determines why and how the resource becomes unavailable based on its transition to a particular state. It also categorizes the resources to differentiate between their graceful and ungraceful transitions to unavailability state. According to the authors, use of resource failure information, unavailability information, checkpoint ability and expected run time of a job can be very useful for grid schedulers for scheduling applications on the grid. The multistate model has five states which are: available to Grid, user present, CPU threshold exceeded, job eviction or graceful shutdown and unavailable. A resource may transit between these states from time to time. A resource will be available if the machine is connected to the network with local CPU load less than the threshold and idle time to be non-zero. It transits to user present state if a connected I/O device like mouse or keyboard is activated. CPU threshold exceeded state is reached when the load of local CPU exceeds the threshold. A transition to job eviction state is performed based if any of the following conditions hold:

a) The resource is suspended for a long time.

b) The machine is evicted during its execution

c) The machine shuts down

Lastly, if a machine is unreachable or fails, a direct transition to unavailable state is made. Based on this state model, unavailability types are classified as graceful and ungraceful. A graceful transition is one when a job enters a job eviction state during its execution or suspension, whereas

ISBN: 978-0-9957075-1-1

ungraceful transition occurs when a direct transition is made to unavailable state.

Further enhancement had been proposed to forecast multi-state availability of a resource in [14]. The study shows that analyzing a resources behavior with its history can be a good predictor. The accuracy is enhanced through transition weighting schemes. To decide as when to assess the resource behavior, the predictor uses two approaches. First is to examine the availability behavior for the previous days and the second approach uses the most recent hours of activity preceding the prediction.

The authors in [16] propose an algorithm using transition N-day with equal transition weight is presented in [14]. The algorithm is assumed to have a centralized job scheduler. All jobs are submitted to the scheduler which then forwards them into a job queue. The job scheduler should know in advance the required execution time by each job. After execution the results are sent to the users who submitted the jobs. The scheduler sends the jobs into a queue and uses First Come First Serve for sorting the jobs. If jobs are present in the queue and resources are available then the scheduler searches for the resource by using the TDE prediction method [14].The purpose of using this method is to determine the reliability of idle resource. A threshold is set for the resource availability. If the resource is suitable and its reliability is above the specified threshold, the job will be assigned to that resource else alternative available resource need to be found by the scheduler. In case of unavailability of a suitable resource, the job will be held by the scheduler.

B. Rule Based

Such approaches handle prediction by deriving useful rules in order to search relevant resources and then assigning tasks to those available resources.

1) Rough Set Theory

The study in [3] uses rough set analysis to efficiently predict a node's behavior. An on online announcer approach is used to determine available resources at present or for future thereby eliminating the need for an inquiry from the grid scheduler. The authors intend to provide solution for resource discovery and task assignment. A new algorithm is proposed in cooperation with rough set tool. The main goal of using rough set is generation of useful rules in order to search for relevant nodes and then assigning tasks to the nodes which are available. The algorithm takes Nodes Information Data Table as input and generate appropriate rules as output. Three attributes (start time, final status of task and completion time) are used as decision attributes. Out of these three attributes, when one of the attribute is used as the decision attribute then the remaining two can be used as conditional attributes depending upon the requirements.

C. Machine Learning

Approaches under this category performs prediction by learning from data. The algorithms build a model based on inputs and then use that to make predictions. Pattern Recognition and classification under machine learning performs pattern matching. It does so by looking at recent resource availability patterns and then examine for similar trace from the past.

The study in [1] focuses on prediction of machine availability by using the techniques of Baysian methods and Support Vector Machines. Another contribution is the use of a time series framework for the automation of correlation search and selection of attributes which lead to efficient prediction. The authors also mention that availability and user login behavior are important characteristics which can be predicted in such (desktop pool) environments efficiently and accurately. The important contributions of the authors are towards the:

1. Automation of the process of finding correlations between traces.
2. Selection of predictive attributes in an efficient way.
3. Usage of walk forward evaluation techniques and overfitting elimination.
4. Switching between various classifications algorithms based on accuracy and effective demands.

1) Pattern Recognition & Classification

The authors in [8, 9] presented the use of pattern matching technique for resource availability prediction. The predictions are done in two dimensions. One of them as instance based where the availability is determined and the other is duration based which involves making predictions on a specific duration. The Austrian grid data trace is examined for pattern matching purpose. The resources were classified into 3 groups as dedicated, temporal and on demand based on their availability.

To rank grid resources and provide optimized resource selection, static and dynamic comparison of resources is provided by using various metrics. For static comparison among resources Mean Time Between Failure (MTBF) and Mean Time to Reboot (MTR) were used whereas for dynamic comparison resource stability and dependability measures were taken. The authors suggested metrics are different and novel from others as previously the comparisons were made by considering only daily and hourly resource availability patterns which did not prove to be much accurate. After observing various availability patterns, two classes were identified. The first one as having lower availability at weekends and higher availability during working week days and second as vice versa. Boyer Moore string matching algorithm was used for pattern matching which showed good performance than others. The major contribution was to compare resources based not only on their daily and hourly availability but also according to MTBF, MTR dependability for different jobs.

2) Lazy Learning /Instance Based Learning

A study of resource availability prediction in Enterprise resource Grids is presented in [10]. The authors proposed to

ISBN: 978-0-9957075-1-1

use Jacard Index using lazy learning to predict the resource availability. The technique is applied on the data trace taken from Microsoft Corporate Network and Planet Lab test bed.

The trace is divided into two data sets labeled as training and test sets respectively, each comprising of binary values of 0 or 1 which indicates non-availability and availability of resource respectively. The training data set is split into two windows of same size named as window furthest and window recent. A Jacard Index (JI) is computed by taking the ratio of number of shared attributes between the two sets of data. After calculating JI for the window furthest, the prediction availability value of that window is selected which will be the next data item after the window ends. The window is then slided next and the JI is calculated for the next window furthest along with the availability prediction of that window. This process continues till the window slides to the second last element of the training data set. This technique uses a fixed window size of 3 elements. In case, widow size is dynamic then the prediction value will also show variations. So, to handle this scenario, the authors proposed to apply majority voting to generate a single prediction value for windows with different sizes but having same JI. The study in [6] address the challenge of predicting the response time of application executing on the grid. According to the authors, response time of an application can be used to identify the availability of a resource in an Enterprise Desktop Grid Computing (EDGC). To improve prediction accuracy, authors suggest to use the state of a resource. Two prediction techniques are presented which are Statistical-Instance Based Learning (S-IBL) and Slow Down of an Application based on Processing and Networking Performance (SdPN). S-IBL is used to consider the state (load conditions) of the resource based on statistical data mining. SdPN is used to dene an applications slow down based on network and resource performance. A simulation engine is used to define that slow down. Additionally a Self-Adjustable Correction Model (SAC) is developed. S-IBL uses past experience for deriving predictions. An experience has input and output features and refers to something happened in the past. Input features identify the conditions under which an experience was occurred and output features refers to the results under those conditions. A data base of all experiences is made. A query is then provided to the data base to match with input features to predict the estimated output features. The query is resolved by determining its similarity with the experiences in the database. The experiences showing relevance to the query, their output features are used to predict query output features.

D. Time Series

These methods in time series analysis are used to extract useful statistics from the data. Models are used for the prediction of future values based on previously observed values. The model performs prediction by first learning which includes resource categorization based on past history and then make predictions for the availability of a resource at a particular time interval.

A Local User Pattern Analyzer (LUPA) architecture is proposed in [5] which has three subsystems namely, data collection, pattern analyzer and predictor. Data collection gathers CPU and RAM usage for every five minutes. Pattern analyzer implements clustering and predictor is responsible for run time predictions based on resource usage. This module is initialized on each machine to allow its resources to be used in the grid. For this architecture, Usage Pattern Analysis (UPA) was used to discover the local usage pattern of a resource. To discover patterns of use, cluster analysis is done on the resources past record. UPA considers resource usage as an object which is a vector of values representing time series of machines resource use. The main resources considered for machine allocation decisions are CPU use and RAM availability. The process is divided into learning and prediction phases. The learning phase categorizes resource use based on recent history. This is done by collecting large amount of data objects. Fixed numbers of Clusters (k) are made from this data. A prototypical object is calculated for each cluster which represent the class. The output will be k prototypical vectors. UPA requires many parameters which are: no. of clusters, data normalization, and computation of prototypical element, clustering algorithm and similarity measure.

E. Data Mining

Clustering and classification are the main prediction approaches under this category. Resource availability is determined by first making groups (clusters) of the availability patterns according to the common resource usage. This information is then used by the classifier and then predictions of resource future availability are made by supplying test data to the classifier. A study to determine and assess predictive methods for ensuring resource availability is presented in [2]. The main goal is to predict that a number of N hosts will be available for a time T. The authors also focus on determining the factors affecting prediction error and determining resource predictability indicators. Naive Bayes and Decision trees predictive model were used for prediction purposes. A sample in training and test data is represented as a binary string (01) as it refers to one hour duration. A prediction is computed at time T. The author consider the interval [T,T+p] to determine the complete availability against partial or complete unavailability where p refers to prediction interval length and its values are [1,2]. The prediction accuracy is quantified by a ratio called prediction error. The prediction interval length strongly affects the prediction accuracy.

The study of resource availability prediction in P2P network in [11] showed that the major problem is lack of central server management for keeping track of available or assigned resources. Therefore, the authors proposed an architecture for resource selection by using group availability data. The main idea is to classify the resources in groups (based on their common usage patterns) and then predicting their availability.

ISBN: 978-0-9957075-1-1

F. Function Approximation

These techniques simulate a target value, which is expected to be the output of an unknown function of measured system variables as input data.

1) Stochastic Model

A computing system using grouping of resources to achieve reliability is proposed in [15]. The study presents the idea of grouping resources for availability in grid. In literature, to provide reliability for application execution, a replicated execution model is adopted in which one version of application executes on primary resource and the other is executed on back up or standby resource. The authors provided a comparison of their proposed computing model having resource repairing facility with the existing replicated model for resource reliability and availability. The availability of both computing systems is modeled using Markov Model. As there is a probability of failure of both primary and back up resources, this may lead to a failure of application execution. As an alternative, the application may be scheduled to execute on multiple grid resources in parallel where all of them will be considered as primary. If the execution completes on one of the resources, it will be considered complete. The problem with this approach is underutilization of grid resources, as majority of resources are kept to execute the same application. Additionally, in such replicated systems, no mechanism of resource repairing is provided. The resource might degrade or become unavailable with the duration of application execution.

So, for reliable execution of application in the grid [15] presents a computing system with resource repairing to ensure application execution. The repairing is done by replacing failed resources with matching available resources. Initially all resources are made available in the grid. A resource manager is responsible to manage the resources by receiving resource request and then matching the requested resources. Resources are organized in a primary back up manner for application execution. A primary resource executes the application and the other resources act as back up. During application execution, resources transit between the states of 1(available) and 0(unavailable). If a primary resource becomes unavailable, the application is migrated to the backup resource, which is then designated as primary. Application execution is not affected if a backup resource turns unavailable. The repairing of a resource is done by replacing failed resources with matching available resources.

III. CONCLUSION

This survey paper was an attempt to discuss in detail the various approaches and methods to address the challenge of resource availability prediction in the grids. An extensive review of various techniques with their accuracy levels along with their pros and cons has been shown in Table.1. Additionally, the techniques are classified according to a taxonomy as exhibited in Figure.1.

REFERENCES

[1] Artur Andrzejak, Patricio Domingues, and Luis Silva. Predicting machine availabilities in desktop pools. In Network Operations and Management Symposium,2006. NOMS 2006. 10th IEEE/IFIP, pages 14. IEEE, 2006.

[2] Artur Andrzejak, Derrick Kondo, and David P Anderson. Ensuring collective availability in volatile resource pools via forecasting. In Managing Large-Scale Service Deployment, pages 149161. Springer, 2008.

[3] Asgarali Bouyer, Ehsan Mohebi, and Abdul Hanan Abdullah. Using self announcer approach for resource availability detection in grid environment. In Computing in the Global Information Technology, 2009. ICCGI'09. Fourth International Multi-Conference on, pages 151156. IEEE, 2009.

[4] John Brevik, Daniel Nurmi, and Richard Wolski. Automatic methods for predicting machine availability in desktop grid and peer-to-peer systems. In Cluster Computing and the Grid, 2004. CCGrid 2004. IEEE International Symposium on, pages 190199. IEEE, 2004.

[5] Marcelo Finger, Germano C Bezerra, and Danilo R Conde. Resource use pattern analysis for predicting resource availability in opportunistic grids. Concurrency and Computation: Practice and Experience, 22(3):295313, 2010.

[6] Josep L Lerida, Francesc Solsona, Pordio Hernandez, Francesc Gine, Mauricio Hanzich, and Josep Conde. State-based predictions with self-correction on enterprise desktop grid environments. Journal of Parallel and Distributed Computing, 73(6):777789, 2013.

[7] James W Mickens and Brian D Noble. Exploiting availability prediction in distributed systems. Ann Arbor, 1001:48103, 2006.

[8] Farrukh Nadeem, Radu Prodan, and Thomas Fahringer. Characterizing, modeling and predicting dynamic resource availability in a large scale multi-purpose grid. In Cluster Computing and the Grid, 2008. CCGRID'08. 8th IEEE Inter-national Symposium on, pages 348357. IEEE, 2008.

[9] Farrukh Nadeem, Radu Prodan, Thomas Fahringer, and Vincent Keller. Availability-based resource selection risk analysis in the grid. Technical report, CoreGRID Technical Report, Number TR-0169, 2008.

[10] Mustazur Rahman, Md Raul Hassan, and Rajkumar Buyya. Jaccard index based availability prediction in enterprise grids. Procedia Computer Science, 1 (1):27072716, 2010.

[11] Karthick Ramachandran, Hanan Lutyya, and Mark Perry. Decentralized approach to resource availability prediction using group availability in a p2p desktop grid. Future Generation Computer Systems, 28(6):854860, 2012.

[12] Xiaojuan Ren, Seyong Lee, Rudolf Eigenmann, and Saurabh Bagchi. Prediction of resource availability in ne-grained cycle sharing systems empirical evaluation. Journal of Grid Computing, 5(2):173195, 2007.

[13] Brent Rood and Michael J Lewis. Multi-state grid resource availability characterization. In Proceedings of the 8th IEEE/ACM International Conference on Grid Computing, pages 4249. IEEE Computer Society, 2007.

[14] Brent Rood and Michael J Lewis. Resource availability prediction for improved grid scheduling. In eScience, 2008. eScience'08. IEEE Fourth International Con- ference on, pages 711718. IEEE, 2008.

[15] Major Singh and Lakhwinder Kaur. Resource grouping in grid environment towards the availability and reliability of computing service. Journal of Advanced Computing, 1:18, 2013.

[16] Jun Zhang and Chris Phillips. Job-scheduling with resource availability prediction for volunteer-based grid computing. In London Communications Sympo- sium, LCS. Citeseer, 2009.

[17] Hu Zhoujun, Hu Zhigang, and Liu Zhenhua. Resource availability evaluation in service grid environment. In Asia-Pacic Service Computing Conference, The 2nd IEEE, pages 232238. IEEE, 2007.

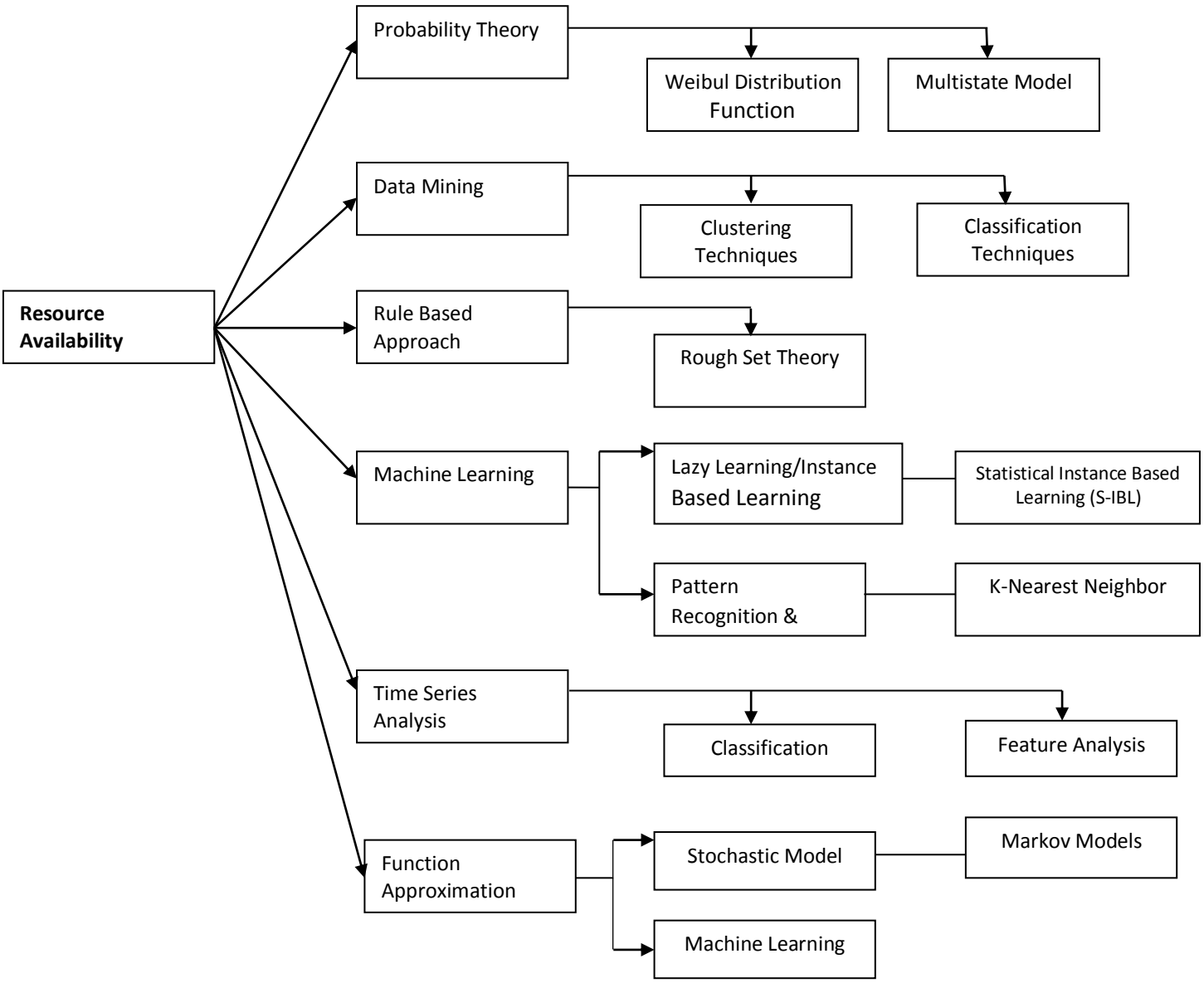


Fig.1. Taxonomy For Classification of Resource Availability Prediction Techniques in Grid

ISBN: 978-0-9957075-1-1

TABLE 1. A COMPARISON OF RESOURCE AVAILABILITY PREDICTION TECHNIQUES IN GRIDS

No.	Year	Prediction Method/Technique	Techniques Taxanomy	Prediction Accuracy	Pros	Cons
1	[6] 2013	S-IBL (Statistical Instance Based Learning)	Statistical /Data Mining	35% increase in prediction accuracy	Potential to adjust to changes in system load	Higher cost of Hybrid Model
2	[15] 2013	Markov Model	Machine Learning	Proposed system showed stable reliability for different application	1. It achieved steady state availability for longer durations 2. Resource repairing feature	None
3	[11] 2012	Resource Usage Pattern	Probability Theory/Data Mining Scheduling Algo's	Accuracy in terms of timings comparison (like availibility for hours/days etc)	Peer 2 Peer grid studies of non-dedicated machines	Human Intervention to setup the infrastructure
4	[5] 2010	Use Pattern Analysis (UPA)	Pattern Classification, Machine Learning	UPA showed 75.6%	Use of Usage Pattern Analysis	It is preferred if data collection of more than 21 days is available
5	[10] 2010	Jacard Index Calculation by utilizing Lazy Learning ,Majority Voting	Artificial Intelligence	96.87% for Microsoft data and 99.74% for PlanetLab data	1. Jaccard Index 2. System is adaptable to the newly arrived data	If window size is dynamic, it assumes that each window has same JI and then calculates the Prediction value based on majority voting.
6	[3] 2009	Rough set approach	Data Mining	98.30%	Resource availability detection based on online-announcer with out any inquiry from Grid scheduler	No real data used
7	[16] 2009	Transitional N-Day (Equal transition weights predictor)	Multi-state model (Probability model)	Depends on T (Resource Reliability Threshold) if T is <50% then the accuracy is approx 73%.	New job schduling algorithm based onTransition N day with equal Transition Day(TDE)	Including more history data for days donot provide better prediction
8	[14] 2008	multi-state availability; simple probability,Transitional N-Day with Equal Transition Weights (TDE) & The Freshness Weighting Scheme (TRF)	Data Mining	78.3% for TDE & 77.3% for TRF	New multi-state availability prediction algorithms	Including more history data for days/hours decreases the performance
9	[8] 2008	Bayes' Rule , Nearest Neighbour	Pattern Recognition and Classification	90% (Instance Based), 70% (Duration Based)	1.Instance & duration based prediction 2.Static & Dynamic comparison of resources	Accuracy decreases if more distant historical data is included
10	[2] 2008	Naïve Bayes & Decision Tree	Data Mining/Machine Learning	Evaluates Success rate in terms of redundancy & no. of hosts, redundancy of 35% can achieve success rate of 95%	1. Investigating factors influencing prediction error like amount of training data, host type & prediction interval length 2.Determining indicators for resource predictability	None
11	[13] 2007	Resource availability states	Multistate Model	Measured in terms of % of time when predictor accurately estimates the next particular state	failure-aware predictive grid scheduling	Linear decrease in accuracy with increase in duration of interval
12	[17] 2007	Strange Theorems	Probability Theory	78%-97% for each metric	Fast Prediction algorithm based on Probability Theory	No. of used parameters is low

ISBN: 978-0-9957075-1-1

13	[12] 2007	Semi Markov Model	Data Mining	86.50%	Proactive management of guest jobs with improved response times	1. Slightly worse predictions on Smaller Window size. 2.Unadaptability of linear time series models for long term predictions
14	[1] 2006	Bayesian Methods, Support Vector Machines	Machine Learning	Mean Squared Error(mse) is the evaluation criteria and shown good results	No power on/off policy makes prediction easy	The machines had been used independently by the users, so no inter-machine correlations were used in predictions
15	[7] 2006	Right Now Predictor, State based predictor, Hybrid Predictors etc	Signal Analysis & Information Theory	PlanetLab accuracy=95% and Microsoft accuracy=87.0%	Use of predictors for tracking uptime state per node	None
16	[12] 2006	Semi Markov Model	Machine Learning	>86.5%	Development of Multistate model by applying SMP for the prediction of temporal reliability	1. Slightly worse predictions on Smaller Window size. 2.Need for deciding a suitable training data set size
17	[4] 2004	Weibull Distribution Function	Probability Theory	95%	Automatic methods to forecast machine availability in peer to peer systems and desktop Grids	Results are useful only for specific application domain

International Journal of Engineering and Applied Computer Science (IJEACS)

Importance of Testing in SDLC

Tanu Jindal

Department of Computer Science & Engineering
Noida Institute of Engineering & Technology (NIET)
Greater Noida, India.

***Abstract*— From previous year researches, it is concluded that testing is playing a vital role in the development of the software product. As, software testing is a single approach to assure the quality of the software so most of the development efforts are put on the software testing. But software testing is an expensive process and consumes a lot of time. So, testing should be start as early as possible in the development to control the money and time problems. Even, testing should be performed at every step in the software development life cycle (SDLC) which is a structured approach used in the development of the software product. Software testing is a tradeoff between budget, time and quality. Now a day, testing becomes a very important activity in terms of exposure, security, performance and usability. Hence, software testing faces a collection of challenges.**

Keywords — Software Engineering; SDLC; Software testing; Verification and Validation.

I. INTRODUCTION

Everyone knows the importance of computer in his life. In today's world computer is using in many fields like industry, education, transportation, medical, agriculture and research. Means, it is becoming an important element in the industry and advanced technology as well as developing countries. In today's life every field is dependent on computer for the betterment of their work. Also, with the help of computer a lot of time is saved. And time saving is indirectly a cost benefit approach. So software engineering is a best approach to develop a computer based system for every field in order to save the time and reduces its cost. Software Engineering is used to develop a system in a systematic way. For this purpose software/system development life cycle (SDLC) is used, as it is the process of developing the system with proper analysis, design, implementation and maintenance to improve the quality of the system. Even SDLC is a systematic approach for the development of the efficient system but without testing it is not possible. Because SDLC tells the process for the development of the system to improve the quality but doesn't helps in finding the defects of the system. So, testing plays an important role in software engineering.

II. RELATED WORK

Gelperin and Hetzel [4] presented the evolution of software test engineering which traced by examining changes in the testing process model and the level of professionalism over the years. Two phase models such as the demonstration and destruction models and two life cycle models such as the evolution and prevention models are given to describe the growth of software testing. Hamlet and taylor [10] presented more extensive simulations, and reach at more precise results about the relationship between partition probability, failure rate, and effectiveness. Vishwas Massey and K.J.Satao [7] in their paper have also compared various SDLC Models for performance and have also proposed a new model for better performance. But both the papers do not make a comparison between the research methodology and SDLC process. Richardson and Malley[1] proposed one of the earliest approaches focusing on utilizing specifications in selecting test cases. They proposed approaches to specification-based testing by extending a wide range of implementation-based testing techniques to be applicable to formal specification languages and determine these approaches for the Anna and Larch specification languages. Madeyski Lech et al.[9] presented the concept of using a set of second order mutants by applying them to large open source software with number of different algorithms. They show that second order mutation techniques can significantly improve the efficiency of mutation testing at a cost in the testing strength. Ntafos [2] presented the comparisons of random testing, partition testing and proportional partition testing. The author guaranteeing that partition testing has at least as high a probability of detecting a failure comes at the expense of decreasing its relative advantage over random testing. Juristo et al. [6] analyzed the maturity level of the knowledge about testing techniques. For this, they examined existing empirical studies about testing techniques. According to knowledge, they classified the testing techniques and choose parameters to compare them. J. A. Whittaker[3] presented a four phase approach to determine how bugs escape from testing. They offer testers to a group related problems that they can solve

ISBN: 978-0-9957075-1-1

during each phase. Claessen et al. [5] developed a lightweight and easy to use tool named "quickCheck", that is a combination of two old techniques (specifications as oracle and random testing) works extremely well for Haskell program. They present a number of case studies, in that the tool was successfully used and also point out some pitfalls to avoid. Harrold et al. [8] presented a new approach to class testing that supports data flow testing for data flow interaction in a class. They also describe class testing and the application of dataflow testing to class.

III. Functions And Goals Of Testing

The primary function of testing is to detect bugs. The scope of testing includes execution of that code in various environments and also to examine the aspects of the code - does the software do what it is supposed to do and function according to the specifications? Testing is an activity that is performed for evaluating software quality and also for improving it. The goal of testing is systematically and stepwise detection of different classes of errors within a minimum amount of time and also with a much less amount of effort. The basic purpose of testing is verification and validation in order to find various errors and problems – and the aim of finding those problems is to get them fixed. Testing is more than just error detection. Testing is done under controlled conditions.

Verification: To verify if system behaves as specified. It is the checking and testing of items, which includes software, for conformance and consistency of software by evaluating the results against pre-defined requirements. In verification we ask a question, are we building the product right?

Validation: In this we check the system correctness which is the process of checking that what has been specified by user and what the user actually wanted. In validation we ask a question: Are we building the right system?

IV. SDLC (Software Development Life Cycle)

SDLC serves as a guide to the project and provides a flexible and consistent medium to accommodate changes and perform the project to meet the client's objectives. There are various stages used in the life cycle of software development. Software development life cycle is basically a systematic way of developing software. It includes various phases starting from the functional requirement of software (means what software is supposed to do). After that designing takes place then development and then testing. After testing is finished, the source code is generally released for Unit Acceptance Testing (UAT) in client testing environment. After approval from client, the source code is released into production environment [11]. There is various software development approaches defined and designed which are used during development process of software, these approaches are also referred as "Software Development Process Models". Each process model follows a particular life cycle in order to ensure success in process of software development. SDLC phases define key schedule and delivery points which ensure timely and correct delivery to the client within budget and other constraints and project requirements. SDLC co-operates project control and management activities as they must be introduced within each phase of SDLC.

There is various software development approaches defined and designed which are used during development process of software, these approaches are also referred as "Software Development Process Models".

V. Phases In Software Development Lifecycle

Testing phase has much importance in SDLC due to a major role in debugging and error correction. The phases of SDLC is being followed in both testing and development cycle of any software application. Here are the phases of SDLC that is being followed:

a) *Requirements Gathering and Analysis:* Under this phase, proper requirements of project are gathered. All close functions are brought in to focus. All kinds of requirements and analysis of user requirement are done in this phase.

b) *System Design:* This is the next phase in SDLC where a rough system design is made. With all data and information being gathered, a system design is made.

c) *Development:* This is the next phase after system design when development of project is made. According to design, proper coding is done to gain that design. Programming language might be selected according to the project.

d) *System Testing:* Just after development phase, testing is carried out to know the outcome of application. Testing is made to know the actual result and the expected result.

e) *Operations and Maintenance:* This is the final stage of SDLC, where the software that is being developed is being distributed to end users who are responsible for maintaining and using it for proper operations. The software that is being developed must be open to any changes being made in coding.

VI. Role Of Testing In SDLC

Testing is required to remove the discrepancy in the software product development process. In order to implement any software product, it has passed through a set of various phases. With the help of testing we can catch small problems before they become big problems later on. Testing activities also provides the chance to review requirements for important quality attributes, to ask questions and to resolve issues earlier. There are many activities that are performed during the testing. These are:

- Test Analysis
- Test Design
- Test Execution

These activities are required to reduce the rework which results in reducing the cost and time. Software testing is an

ISBN: 978-0-9957075-1-1

ongoing process which we can't stop in between. Testing is required in SDLC due to the following reasons:

- To identify the errors
- To remove ambiguity
- To improve the reputation of the company
- To improve quality of the product
- To remove Hazards
- For verification and validation
- To improve reliability
- To improve cost
- To increase the usability

VII. CONCLUSION

From the above discussion it is clear that without testing it is not possible to implement an effective product. If the product is not effective then it will decrease the quality of the product. So, Testing is essential to improve the quality of the system as well as to the success of the overall effort. Testing is performed by developer end and customer end but it can ensure a performance of the product by predicting its behavior. In this paper, it is concluded that testing should be used in all the phases of SDLC and not in one or two phases. Life cycle of Software development is that type of structure which is imposed on the development process of the software product. As there are different activities involved in SDLC so, testing plays a different role in different-different phase. For timely readiness of the system testing is very important as it provides the visibility of the quality of the product at each step.

REFERENCES

[1] D. Richardson, O. O'Malley and C. Tittle, "Approaches to specification-based testing", ACM SIGSOFT Software Engineering Notes, Volume 14 , Issue 9, 1989, pp. 86 – 96.

[2] Ntafos Simeon C. "On comparisons of random, partition, and proportional partition testing." Software Engineering, IEEE Transactions on 27.10 (2001): 949-960.

[3] J. A. Whittaker, "What is Software Testing? And Why Is It So Hard?" IEEE Software, January 2000, pp. 70-79.

[4] D. Gelperin and B. Hetzel, "The Growth of Software Testing", Communications of the ACM, Volume 31 Issue 6, June 1988, pp. 687-695.

[5] Claessen Koen, and John Hughes. "QuickCheck: a lightweight tool for random testing of Haskell programs."

[6] Juristo Natalia, Ana M. Moreno, and Sira Vegas. "Reviewing 25 years of testing technique experiments." Empirical Software Engineering 9.1-2 (2004): 7-44.

[7] Glenford J. Myers, "The Art of Software Testing, Second Edition" Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

[8] Harrold Mary Jean, and Gregg Rothermel. "Performing data flow testing on classes." ACM SIGSOFT Software Engineering Notes. Vol. 19. No. 5. ACM, 1994.

[9] Madeyski Lech et al. "Overcoming the Equivalent Mutant Problem: A Systematic Literature Review and a Comparative Experiment of Second Order Mutation." (2013): 1-1.[LR4]

[10] Hamlet Dick, and Ross Taylor. "Partition testing does not inspire confidence (program testing)." IEEE Transactions on Software Engineering 16.12 (1990): 1402-1411.

[11] Accessibility Summit. (2006). Public Sector NeedsBetter Guidance On Web Accessibility, E-Government Bulletin (Issue 226, 13 November 2006)http://www.ukoln.ac.uk/webfocus/events/meetings/accessibility-summit-2006-11/egovernment- 2006-11-13.php (Accessed August 30th2007)

AUTHOR PROFILE

Tanu Jindal is working as Assistant Professor in Computer Science & Engineering Department of Noida Institute of Engineering and Technology, Greater Noida. She received Master's degree in Software Engineering from Thapar University, Patiala. She is pursuing Ph.D. in Software Engineering. Her main research interests are Software Engineering and Grid Computing. She has guided many M.Tech. students for their thesis. She has guided many B.Tech students in their Projects. She has more than 10 publications in National and International Journals and Conferences.

Call for Papers

International Journal of Engineering and Applied Computer Science (IJEACS) invites authors to submit their manuscripts for categories like research papers, review articles, survey papers, technical reports , report of unsuccessful research projects , case studies , tutorials , book reviews, short communications and cross talk, extended version of conference papers for publication in our monthly issue.

IJEACS reviews manuscripts through double blind peer review process. Authors can submit their original, unpublished manuscript which is not under consideration for publication in any journal or conference through email. IJEACS publish papers in major streams of Engineering and Computer Science include following but not limited to.

Computer Science

- Software Design and Modeling
- Service Oriented Architecture
- Open Source Software
- Software Testing and Maintenance
- Software Measurement and Reliability
- Knowledge based Systems
- Image Processing and Computer Graphics
- Extreme Programming and Formal Methods
- Artificial Intelligence, Image Recognition and Bio metrics.
- Machine Learning and Computer Vision
- Algorithm Analysis and Design
- Computational Mathematics
- Data Structures and Graph Theory
- Video Coding and Data Compression
- Database Systems and Big Data Analytics
- Internet of Things, Architecture and Computing
- Parallel and Distributed Computing
- Cloud Computing, Agent Oriented System

- Communication Network and Systems
- Embedded Systems and Applications
- Cryptography and Information Assurance
- Computational Biology and Bioinformatics
- Human Computer Interaction
- Natural Language Processing

Engineering

- Micro Processor Architecture and Design
- VLSI/IC Microelectronics and Computer Design
- Parallel Processing and Digital Systems
- Wireless Transmission and Mobile Communication
- Antenna and Wave Propagation
- Semiconductors, Circuits and Signal System
- Material Science and Metallurgy
- Machine and System Design
- Robotics and Automated Control
- Manufacturing Processes and CAD/CAM
- Quality Control and Assurance
- Digital Signal Processing
- Photonics, Fiber Optics and Optical Communication
- Biosensors, Electrical-based Interactions
- Nano electronic Devices and Silicon Micro/Nano Electronics Applications
- Distributed monitoring systems and Smart Systems
- Fluid Dynamics and Implementations
- Mechanics and Vibration
- Heat Transfer
- Combustion Engines and Automobiles
- Health Instrumentations and Technologies
- Thermal Engineering
- Solar Power Systems
- Ergonomics

Submit your manuscript to: **submit@ijeacs.com** **or** **ed.manager@ijeacs.com**

Innovations continue to serve the humanity

Issue Highlights

- **Malware**
  Ajit Kumar, K.S. Kuppusamy , G. Aghila
- **Health Care System**
  Mahboobeh Abdoos
- **Cloud Computing**
  Kamalinder Kaur, Nupur
- **Grid Computing**
  Farrukh Nadeem , Mahreen Nasir
- **Software Development**
  Tanu Jindal

ISBN-13:978-0-9957075-1-1
9 780995 707511 >

International Journal of Engineering and Applied Computer Science
Volume: 01, Issue: 02, December 2016
ISBN: 978-0-9957075-1-1