Polarized Gossiping for Enhanced Publish/Subscribe Services over the Internet

Mr.V.Muni **Dr.S.Vasundra** **Mr.C.Raghavendra**

Abstract ***---The publish/subscribe paradigm is an appealing solution as messaging middleware because it offers the time, space, and synchronization decoupling properties that distributed applications such as online gaming, messaging, social networking, and business intelligence require. These applications are typically characterized by a large number of participants scattered across the world that communicate by exchanging messages on a wide area network (WAN), such as the Internet. However, while this kind of middleware fits the generic asynchrony and scalability requirements of large scale complex critical infrastructures (LCCIs), like air traffic control (ATC) systems, it completely or partially lacks the support of quality-of-service (QoS) guarantees. The Proposed System includes P-Coding: a light weight security applied to network coding and a mechanism to choose gossip partners during the recovery phase of the protocol. It uses a polarized gossip, in which a subscriber assigns a weight to a subset of other subscribers. The system has two different methods for assigning the weight. First one is based on the current network status. The second is based on the position of the nodes in the overlay network.***

Index terms: *publish, subscribe, gossiping, p-coding, network coding, key perturbing.*

I. INTRODUCTION

With the increase in the rate of internet users in now a days there is a need for new technologies to emerge. The features of the internet significantly change the degree of connectivity of the users with it, who are located at different networks such as LANs, WANs and Large scale Complex Critical Infrastructures (LCCIs) such as Air Traffic Control Systems (ATC) which are connected through the Internet. The LCCIs are the systems which are involved in exchange of a large amount of messages, which require the dynamic properties like synchronization decoupling. This demand motivates the requirements pertaining to the designing of a new flexible loosely coupled transmission infrastructures that achieve the dynamic decoupling properties. The Publish / Subscribe prototype proved itself as an effective solution for this type of system requirements, where a large scale of events/messages was exchanged. Publish/Subscribe is the best suitable messaging middleware for such loosely coupled transmissions. The potential of an event based transmission is actually attracted by fully decoupling the time, space and synchrony (i.e. the flow of messages between publisher and subscriber).

Some of the challenges that are faced in the earlier systems implementing publish/subscribe paradigm are best effort delivery, message ordering, delivery of message in bounded time and reliability of transmission [3]. The traditional publish/subscribe systems lack the Quality of Service guarantee partially or completely by focus on any one of these constraints or neither of them. For example the applications like social networking and messaging depends on the best effort delivery, some others depend on many other non-functional requirements along with the best effort delivery. Let's consider another example weather forecasting and disaster management system, which collects the weather reports in the form of messages from vast number of stations, scattered over a geographical area and then the main station forwards the reports to concerned authorities to take necessary action if any natural calamity is going to occur. These messages must be delivered within time and complete if not received in time they are useless. So this paper aims to design a mechanism that provides the timeliness and reliability to the publish/subscribe services over the internet.

II. LITERATURE SURVEY

Publish/Subscribe paradigm

The publish and subscribe mechanism is a messaging middleware which fits for the requirements of dynamic time decoupling, space decoupling and the synchrony decoupling [2] in a large scale complex critical system. The node in the overlay network which is creating and publishing an event is called Publisher; other nodes which are receiving that event notification are called Subscribers. The node may be a publisher or subscriber; it depends on the state of the node either receiving or sending the event at that instant. Fig. 1 describes the way publish/subscribe mechanism provides the space decoupling by making the abstraction of details of publisher and subscriber from each other, time decoupling; the participants need not to be active at the same time, lastly synchronization decoupling; there is no need for the subscriber to wait for the publisher to produce an event therefore subscriber can get asynchronously notified.

Publish / Subscribe with Network Coding

Network Coding is a Forward Error Correction (FEC) scheme also known as the random linear network coding (RLNC) [4]. Publish and subscribe mechanism on combination with network coding will give more efficient system which makes it more reliable. In network coding the whole message to be transmitted was broken into *n*

ISSN: 2278 – 1323

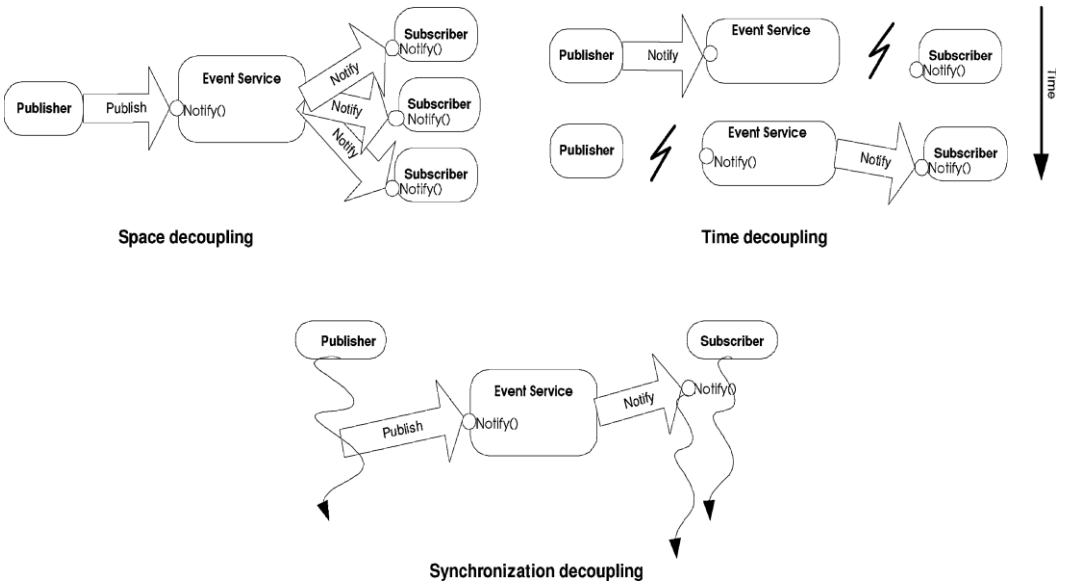

Fig.1. Space, time and synchronization decoupling with Publish/subscribe paradigm

original packets ($p_1, p_2, p_3, ..., p_n$) and transformed into a set of *n* linearly independent combinations and generate redundancy virtually to recover any packet loss by linear combination of original packets. Each of such linear combinations l_i is calculated by the following equation.

$$l_i = \sum_{i=1}^{n} k_i \, p_i$$

Here k_i is a co-efficient taken randomly from an interval 0, 1, … , (q-1) excluding the all zero coefficients. Where all the operations are done over the Galois Field GF(2^w), 2^w=q. with the use of RLNC the reliability property of publish and subscribe prototype will be achieved. The number of gossip rounds to be performed to recover *m* lost packets had drastically fallen from *O(mlog(m))* to *O(m)* . Hence the probability of reconstructing a whole message from a plain packet received increases linearly, in network coding it increases exponentially.

Publish / Subscribe with Gossiping

Gossiping is the so called epidemic approach [5], in which the event/message transmitted like a contiguous disease spread or diffusion of perfume in the air. Whenever the node receives a message it stores that in a buffer of size 'b', called as *fan_in*, if the node forwards a message limited number of times 't' to other nodes in the overlay network is called as *fan_out.* There are three different types of gossiping algorithms are there at present.

(i) *Push*: immediately on receiving a message the node forwards that to other node.

(ii) *Pull*: periodically the identifiers of set of messages received will be sent to other nodes. On recognizing that a message is missed by comparing the set of Ids with local history it makes an explicit pull request for the missed packet.

(iii) *Push/Pull*: the node on successfully receiving message diffuses message Id to others. The other nodes which were not received it will make explicit pull request.

Publish/subscribe based event dissemination services are applied with gossiping efficiently and these algorithms provided the timeliness constraint to publish and subscribe services.

III. RELATED WORK

In this section we study about some of the projects which make use of the publish/subscribe mechanism as major constraint. The earlier implementations of publish/subscribe scheme by several industrial strong solutions were based on idea of topic or subjects (e.g. Talarian Corporation [6]). The topics are like groups in the communication and each topic is identified by a keyword. Some of the first systems offering publish/subscribe systems were based on the ISIS [7] . Besides the improvements like hierarchical addressing and wild cards topic based publish/subscribe scheme reflects the static scheme with limited offerings of expressiveness. The content or property based systems like Rosenblum and wolf [8] improves publish/subscribe scheme by proposing a new subscription mechanism by taking actual content of event as a basis. In other words the subscription is not based on the predefined topic names but properties of the events such as data structures which are carrying the events, like Gryphon [9]. The subscribers may also use meta data associated to the events, JMS[10]. In content based publish/subscribe systems subscribers select events by using filters, which are defined as a name value pairs

with basic relational operators (= ,<,>,≤,≥) to identify valid events. The topics generally again classified and grouped according to the similarities in both the structure and content of the events, this has led to the type based publish/subscribe scheme by replacing name basis with a scheme that filters the

ISSN: 2278 – 1323

events based on their type Eugster [11]. This makes the language and the middleware integrated closer enough. The commercial systems such as DDS [12], and JMS [13] are designed to work efficiently in small and medium networks but its facing many problems and performance degrading gradually in WANs. By studying all these systems this paper presents a new frame work for publish/subscribe services over WAN such as Internet that mainly focus on the reliability, timeliness and security against eves dropping. The first solution which merged network coding and gossiping with publish/subscribe prototype is in that system the RLNC and gossiping are performed at receiver side: the existing system on contrary to this implement RLNC at sender node and gossiping at the receiver node. Some others performed dissemination of plain data by means of push/pull based gossiping algorithm.

IV. PROPOSED SYSTEM

In this paper we proposes a scheme that makes publish and subscribe services enhanced with security and improved reliability, timeliness for the event notification services. Though the previous works achieved the reliability and timeliness [1] it lacks the security and effective recovery. This drawback can be overcome by applying the p-coding scheme to network coded packets to make the system resilient to eves dropping attack and gossiping is polarized to assure the probability of getting a message recovery increases.

P-Coding: It is a light weight encryption mechanism that uses permutation to encrypt the message by scrambling the packets by considering them as individual symbols.

As shown in the Fig. 2 network coded packets along with their Global Encryption Vectors (GEVs) are passed as input to the P-coding then it permutated the messages by considering each packet of every message as a special symbol.

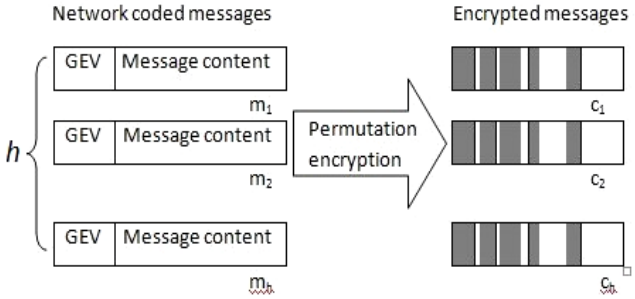

Fig.2. Permutation encryption on network coded messages.

Algorithm for Key Perturbing.

Key_Perturbing(k,n,m,s,d)

1: for each iϵ[1,m-1] /*to generate the sequence $[x_1,\ldots,x_{m-1}]$*/
2: x(i)←d%(i+1) d←d/(i+1);
3: for each iϵ[1,m-1] /* to generate the sequence $[y_1,..,y_{m-1}]$*/
4: y(i)←m-x(m-i);
5: for each iϵ[1,n] /* initialization */
6: π(i)←i;
7: for each iϵ[1,m-1]/*to calculate the m-partial permutation*/
8: π(s-1+i)↔π(s-1+y(i));
9: for each iϵ[1,n] /* to perturb the current key k using π */
10: $k^{-}(i) \leftarrow \pi(k(i))$;
11: return $k^{-}(i)$;

This random key generation algorithm returns a permutation encryption key. There are after the key is send to the receiver for encryption the key changes for every transmission based on the key perturbing algorithm.

Polarized Gossiping: Gossiping is performed at the time of recovering any lost packets is found at receiver/subscriber side. The existing systems implements random uniform selection mechanism for gossip partners, those who are participating in gossip rounds. Because of selecting the participants in such random manner over a large public network like internet there is a high probability of missing the right partner to gossip, so that the recovery of packets will not be successful without the need for another gossip round in most of the cases.

Hence we came up with new approach of polarization. The polarized gossiping implements two techniques to select the right gossip partner. They are

Technique (i) selecting the partner based on the probability of that node having the missed packet.

Technique (ii) selection of partner based on the position of that node in overlay network, i.e nodes near the sender have high probability of successfully receiving a message.

Based on these two techniques a subscriber select another subscriber of same event to recover a lost packet through polarized gossip round.

As shown in the Fig. 3 the encrypted packets are disseminated from a publisher to all the subscribers of that corresponding event published. If any subscriber detects that the message was incomplete then a gossip round is performed to recover complete message. The nodes *s1*and *s2* has enough packets to reconstruct the whole message but *s3* has to go for a polarized gossip round.

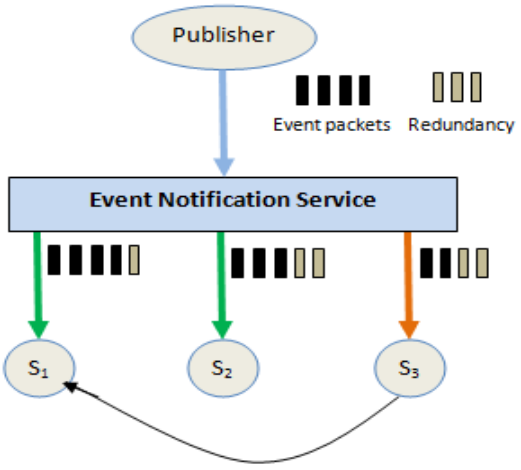

Fig.3. Dissemination and recovery of an event notification.

ISSN: 2278 – 1323

4.1 Node Architecture

Each and every node in the network implements the architecture as shown in the Fig. 4, it comprises of three main blocks.

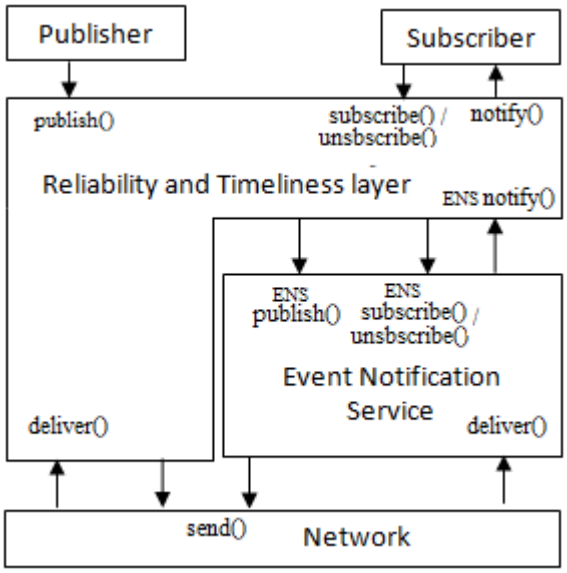

Fig. 4. Architecture implemented by a node in the network.

(i)*Application:* This block has to play two distinct and individual roles. They are; Publisher i.e the information provider, the subscriber who consumes the information provided by the publisher application.
(ii)Reliable and Timeliness Layer: This part of the node is important, where we implement all the techniques to enhance the services of system like security, reliability and timeliness.
(iii)Event Notification Service: The mediator between the publisher and subscriber interface which implements the following services;
publish(): to publish an event in the system, invoked by publisher
subscribe(): to subscribe for the interested events or topics, invoked by subscriber
notify(): to deliver the notification to subscriber, invoked by ENS.
unsubscribe(): to unsubscribe from previously subscribed topic or interest, invoked by subscriber.

4.2 Network model

The network on which we are applying the proposed work is considered as a classical two state Markov model which is invented by Gilbert- Elliott [15]. Through this network model we can easily categorize the patterns of errors occurred in transmission process and also to know the efficiency of the network coding to detect and correct the errors [14]. This model generally follows the indications G for good state and B for bad state of a node in the network. Both the states generate errors as independent events at a dependant state with error rate for good state as (1-k) and for bad state as (1-h). The fig. shows the 2 state Markov chain model. The links among the nodes are not reliable and can be measured [15] by P and Q the probability to loss a packet is called packet loss ratio (PLR) and the mean number of lost packets is called average burst length (ABL).

$$P=\frac{PLR.Q}{1-PLR}\ ,\quad Q=ABL^{-1}$$

P is the probability to transform from G to B,
R is the probability to transform B to G,
(1-p) is the probability of a node to be in good state,
(1-r) is the probability of a node to be in bad state.
To analyze the data loss, we consider an event as arrival of a packet and error as a packet loss. The matrix T represent transition matrix and is having two transitions.

$$T=\begin{bmatrix}1-p & p\\ r & 1-r\end{bmatrix}$$

$$p=P(q_t=B|q_{t-1}=G);\quad r=P(q_t=G|q_{t-1}=B)$$

The probability of stationary state is given by π_G and π_B for 0<p and r<1 from this state the error rate p_E is

$$p_E=(1-k)\,\pi_G+(1-h)\,\pi_B;$$
$$\pi_G=\frac{r}{p+r}\ ,\ \pi_B=\frac{p}{p+r}$$

Based on this probability of a node's stationary state weights were assigned to participate in polarized gossip round

V. SYSTEM EVALUATION

In this paper, we have proposed a strategy that improves coding and gossip for secured, reliable and timely event dissemination over the Internet. We have conducted a theoretical analysis to evaluate the ability of gossip to retrieve missing information in a small number of rounds.

Reliability: The number of received events to the number of published events ratio is called as the success rate, referred to as reliability. Reliability is the capability of publish/subscribe service successfully delivering all the events published to all the subscribers. If all the published events were successfully received by all the corresponding subscribers then the success rate will be '1'.
The coded message transmission and no coding case at different degrees of redundancy clearly says that the number of redundant packets required to recover from faulty transmission is largely higher in no coding case than the case of coding, exhibits an efficient recovery capability and pull gossip strategy is better for recovery.

Overhead: this is the ratio between the number of packets exchanged at the time of transmission to the total number of packets produced by publisher. This measure represents the traffic load and must be low to avoid congestion. In our system overhead caused only with the increasing degree of redundancy but not on the coding strategy. So we made the necessary controlling steps by limiting the fan_in of a node.

Timeliness: Performance of system is measured by mean latency, how fast and efficient the dissemination algorithm is

ISSN: 2278 – 1323

capable of delivering the event notifications .Because of fault tolerance mechanisms the performance may be fluctuating and is measured by standard deviation.

The publish/subscribe services merged with gossip and coding mechanisms for achieving at most reliability by varying redundancy degree. With increase in the value of fan_in the delivery latency also increases there by compromises the performance which effects the timeliness property of the publish/subscribe services. So we make the value of fan_in by default as 1.

Security: The system is provided with the most efficient security algorithms like key perturbing and network coding. Firstly to locate the range of key vales for key perturbing it requires *O(n)* different choices. Then to find the key the attacker has to go through *O(m!)* ways. Finally to crack the network coding, to perform the Gaussian elimination it costs *O(h3)* multiplication operations. Hence, the computational complexity is *O(n·m!·h3)* in terms of multiplication operations means that the system is robust to any attack in the network.

As shown in the graph (Fig.5) the performance of the system is increased with success rate by reduction in the redundancy degree and latency of notification delivery through polarized gossiping and network coding. This changes in performance is achieved by reducing the redundancy by limiting the number of retransmissions required or gossiping rounds by a node in recovery phase through polarized gossiping. Success rate depends on the capability of nodes receiving the notification with polarized gossiping the participants of gossip round were selected with at most probability of having the missed packet recovered, so success rate also known as reliability increases tremendously. The fan_in is set to '1' by default means the redundancy degree is controlled so that the latency is also reduced and giving chance for all the faulty nodes to recover the lost event notifications. Thus overall performance of system is increased.

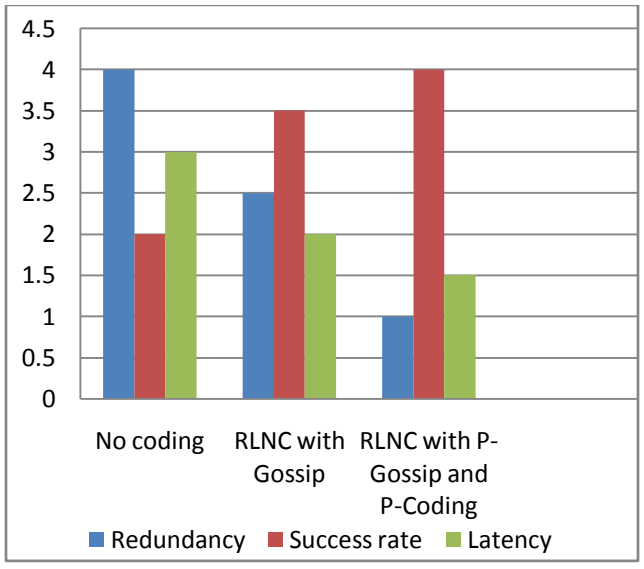

Fig.5 . Performance of the system.

VI. CONCLUSION

In this work we proposed a light weight encryption scheme for providing the security to the system. The coding reduced the number of the retransmissions required there by improves performance of system which achieved the timeliness of event notification services. The obtained results proved that polarized gossiping improves the reliability by lowering the number of nodes to be contacted during a gossip round in recovery and increased the success rate of the nodes in the overlay network achieved the reliability. Here, the p-coding on combination with network coding reduced the vulnerability of system from a loss transmission and ensures secured dissemination of event notifications over the internet.

References:

[1] Christian Esposito, Marco Platania, and Roberto Beraldi, "Reliable and Timely Event Notification for Publish/Subscribe Services Over the Internet" IEEE/ACM Transactions on Networking, Vol 22,No1,February 2014.

[2] P. Eugster, P. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many Faces of Publish/subscribe," ACM Computing Surveys (CSUR), vol. 35, no. 2, pp. 114–131, June 2003.

[3] E. Fidler, H. Jacobsen, G. Li, and S. Mankovski, "The padres distributed publish/subscribe system," in Feature Interactions in Telecommunications and Software Systems, vol. 8, 2005, pp. 12–30.

[4] C. Fragouli, J. L. Boudec, and J. Widmer, "Network coding: an instant primer," Computer Communication Review, vol. 36, no. 1, p. 63, 2006.

[5] A.-M. Kermarrec, L-Massouli´e, and A. J. Ganesh, "Probabilistic Reliable Dissemination in Large-Scale Systems," IEEE Transactions on Parallel and Distributed Systems (TPDS), vol. 14, no. 2, pp. 1–11,February 2003.

[6] Everything you need to know about middleware: Mission-critical interprocess communication. White paper. Talarian Corporation 1999, Los Altos, CA (now part of TIBCO, Palo Alto, CA). Available online at http://www.talarian. com/.

[7] BIRMAN, K., COOPER, R., et el 1990. The Isis System Manual. Dept. of Computer Science, Cornell University, Ithaca, NY.

[8] ROSENBLUM, D. ANDWOLF, A. 1997. A design framework for Internet-scale event observation and notification. In Proceedings of the 6th European Software Engineering Conference/ACM SIGSOFT 5th Symposium on the Foundations of Software Engineering. ACM Press, New York, NY, 344–360.

[9] BANAVAR, G., CHANDRA et.el. 1999a. An efficient multicast protocol for content-based publish-subscribe systems. In *Proceedings of the 19th International Conference on Distributed Computing Systems* (ICDCS'99).

[10] HAPNER, M., BURRIDGE, R., SHARMA, R., FIALLI, J., AND STOUT, K. 2002. *Java Message Service*. Sun Microsystems Inc., Santa Clara, CA.

[11] EUGSTER, P. AND GUERRAOUI, R. 2001. Contentbased publish/subscribe with structural reflection. In *Proceedings of the 6th Usenix Conference on Object-Oriented Technologies and Systems* (COOTS'01).

[12] OMG. (2007, January) Data Distribution Service (DDS) for Real-Time Systems, v1.2. [Online]. Available: www.omg.org

[13] S. Microsystems. (2002, April) Java Message Service, v1.1. [Online]. Available: docs.sun.com/app/docs/doc/816-5904-10

[14] Morgera, S.D., Simard, F.: Parameter estimation for a burst-noise channel. In: International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Washington, DC, USA (1991) 1701–1704

[15] G. Hasslinger and O. Hohlfeld, "The Gilbert-Elliott Model for Packet Loss in Real Time Services in the Internet," Proceedings of the 14th GI/ITG Conference on Measuring, Modelling and Evaluation of Computer and Communication Systems, pp. 1–15, March-April 2008.

ISSN: 2278 – 1323

About the authors:

Mr. V. Muni pursuing M.Tech in software engineering from JNTU College of Engineering, Anantapur, Department of CSE. Completed B.Tech degree from MLIET college, Department of IT. Interested in the fields of operating systems, computer networks.

Dr S. VASUNDRA, presently working as Professor and Head of the Department CSE, JNTUA CEA. She completed her Ph.D from JNTUA University, Anantapur, M.Tech from JNTUA and B.E from VTU. She is having 16 years of teaching experience and 10 years of research experience. Published 20 papers in various international journals and 3 in national journals. Her areas of interest include MANET's, Cloud Computing, Algorithms, Data Structures and Distributed Computing.

Mr. C. Raghavendra currently working as lecturer in Department CSE, JNTUA CEA. He completed B.tech from RGM College of Engneering and Technology, Nandyal and M.Tech from Bharath University, Chennai. Now pursuing Ph.D from Bharath University, Chennai. He is having 6+ years of experience. Published 2 International Journals, Participated in 6 National Workshops and International Workshops, attended 4 National and International conferences. Member in CSTA, ACM and IAENG

ISSN: 2278 – 1323