Privacy-Preservation and location proof updation in PBMS networks

C.Sai Sudha, T.Madhavi Kumari

***Abstract*— Making new connections according to personal preferences is usually a crucial support in mobile online community, where a good initiating user can buy matching end users within actual physical proximity involving him/her. Inside existing methods for this kind of services, usually each of the users specifically publish his or her complete pages for others to locate. However, in most applications, the users' personalized profiles may well contain hypersensitive information that they just don't want to make public. Within this project, we propose FindU, some privacy-preserving user profile matching schemes for proximity-based mobile social(PBMS) support systems. In FindU, an beginning user can buy from a small grouping of users usually the one whose user profile best matches with his/her; to limit danger of level of privacy exposure, only important and minimal information regarding the private attributes from the participating end users is changed. Two increasing amounts of user level of privacy are defined, with decreasing amounts of revealed user profile information. Benefiting secure multi-party working out (SMC) techniques, we suggest novel practices that realize each one of the user level of privacy levels, which can be personalized from the users..**

***Index Terms*— Mobile Networks, Social Networks.**

I. INTRODUCTION

With all the proliferation of mobile phones, mobile support systems (MSNs) are becoming an inseparable a part of our day-to-day lives. Leveraging networked portable devices such as smart telephones and PDAs since platforms, MSN not simply enables people to use their existing online networks (OSNs) with anywhere and anytime, but also introduces a myriad of mobility-oriented purposes, such since location-based providers and augmented reality. One of them, an important service is always to make completely new social connections/friends inside physical proximity using the matching connected with personal profiles. For case in point, are WINDOWS LIVE MESSENGER applications in which match a single with neighbourhood people intended for dating or perhaps friend-making according to common interests. In such an application, a end user only would need to input several (query) attributes in the girl profile along with the system would automatically get the persons around with equivalent profiles.

Manuscript received September 23, 2014.

Sai Sudha persuing M.Tech in the field of Digital systems and computer electronics from Jawaharlal Nehru Technological University,Hyderabad,India.

T.Madhavi Kumari Associate Professor Electronics & Communication Engineering **department and Co-Ordinator ,Academic & Planing of** Jawaharlal Nehru Technological University,Hyderabad,India.

Nevertheless, such methods also raise quite a few privacy considerations. Let us all first analize a pressuring scenario. Within a hospital, patients may include their condition symptoms and medications of their personal profiles in order to find similar sufferers, for bodily or emotional support. With this scenario, an commencing user (initiator) should find out the sufferer having the maximum number connected with identical signs and symptoms with the girl, while currently being reluctant to reveal her hypersensitive illness facts to the rest of the users, along with the same to the users currently being matched with. If users' exclusive profiles are usually directly exchanged together, it will probably facilitate end user profiling exactly where those information is usually easily collected by way of nearby end user, either in the active or perhaps passive approach; and those user information could possibly be exploited with unauthorized means. For case in point, a salesman from the pharmacy may submit destructive matching queries to have statistics in patients' drugs for advertising purposes. To cope with user profiling with MSNs, it is vital to expose minimal and necessary personal information to since few users as you can.

In reality, the best situation is always to let this initiator and best coordinating user right and privately find out and connect together, without learning anything with regards to other users' page attributes, while the rest of the users must also learn nothing around the two user's coordinating attributes. Nevertheless, it is challenging to determine the coordinating users secretly while effectively. One may imagine simply turning journey cell phone or perhaps input not many attributes, but these kind of would obstruct the method usability.

II. PREVIOUS WORK

With the proliferation of mobile devices, mobile social networks (MSNs) are becoming an inseparable part of peoples' lives. Leveraging networked portable devices such as smart phones and PDAs as platforms, MSN not only enables people to use their existing online social networks (OSNs) at anywhere and anytime, but also introduces a myriad of mobility-oriented applications, such as location-based services and augmented reality.

Among them, an important service is to make new social connections/friends within physical proximity based on the matching of personal profiles. For example, MagnetU1 is a MSN application that matches one with nearby people for dating, friend-making or small-talks based on common interests.

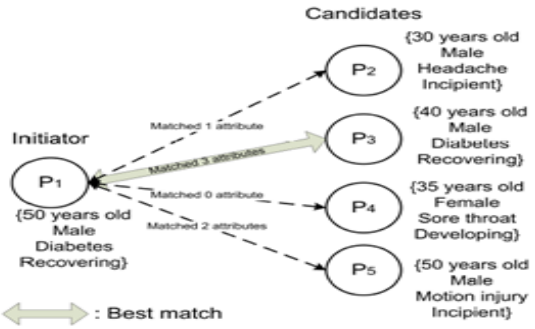

Figure 1: Profile Matching in Social Networks

In such application, a user only needs to input some attributes in her profile, and the system would automatically find the persons around with similar profile. The usage of these applications is very broad, since people can input anything as they want, such as hobbies, phone contacts and places they have been to. The latter can even be used to find "lost connections" and familiar strangers.

However, such systems also raise a number of privacy concerns. Let us first examine two motivating scenarios. In a dancing club, people would list their dating preferences in their profiles, which may also include sexual inclinations such as "gay". An attacker may be interested in finding out exactly the persons having such sensitive attributes (called *individual profiling*). To avoid from being profiled, one may want to reveal his attributes only to an initiating user (initiator) whose query criteria best matches with his profile, e.g., there are 10 identical attributes between them. Also, an initiator would like to disclose her query attributes to as few people as possible. In a hospital, patients may include their disease symptoms and medications in their profiles in order to find similar patients, for physical or mental support. However, a salesman from some pharmacy may submit malicious matching queries to obtain statistics on medications of the patients', for marketing purposes. This gives rise to the *group profiling*.

In fact, the ideal situation is to let the initiator directly find out the person that *best* matches with her interests, without knowing any other user's profile information, while the non-best matching users also learn nothing about the initiator's matching criteria. This is illustrated in Fig. 1, where the initiator *P*1 finds and connects to *P*3 directly and privately. Therefore, to cope with user profiling in MSNs, it is important to disclose as little personal information to as few people as possible. Since it is not desirable to simply publish all the profile attributes, it is challenging to privately find out the users with matching profiles. One may think of simply turning off the cell phone or input very few attributes, but these would interfere with the usability of the system. Recently, Yang *et.al*. proposed E-SmallTalker, the first practical system for matching people's interests before initiating a small-talk. Unfortunately, E-SmallTalker reveals the exact matching attributes between the initiator and every other user. [1]

Even though one may be willing to report his own information, he may not have others' permission to report their information, i.e., their phone IDs. Moreover, a central server can be a performance bottleneck and a single point of failure and has the risk of being compromised. Consequently, a centralized system for small talk would be less likely to reach a critical mass of users than systems without the above roadblocks.

First, Bluetooth is the most suitable communication technology for our purposes. There are mainly four types of communication technologies available on mobile phones: cellular networks, IrDA, Wi-Fi, and Bluetooth. Communicating data via cellular networks is costly and often unreliable in typical social settings, e.g., inside a building. Infrared Data Association (IrDA) is limited to line-of-sight communication within a very short distance (e.g., 1 meter), which may be considered intrusive among strangers. Wi-Fi is only available on relatively high-end mobile phones. In contrast, Bluetooth is available on nearly all mobile phones and its communication range is 10 meters on class II devices. Hence we choose Bluetooth as our communication technology.

However, in order to develop a Bluetooth application on mobile phones, we need to overcome several obstacles. For security reasons, a mobile phone will ask for user permission to initiate or accept a Bluetooth connection as well as a pass code for pairing. Hence, an application that relies on Bluetooth connections to transmit data requires explicit user interactions. This requirement not only requires users to "babysit" the system but is also too intrusive for strangers. Therefore, we need to find a way for two phones to exchange information without establishing a Bluetooth connection.

We achieve this by using Bluetooth Service Discovery Protocol (SDP) to publish/exchange information. In SDP, each service is represented by a service record that is identified by a 128-bit Universally Unique Identifier. All information about a service that an SDP server maintains (on a phone) is contained within a single service record, which consists of a list of service attributes. Each service attribute describes a single characteristic of a service (e.g., its name, type, parameters, protocols used) and consists of an attribute ID and the corresponding attribute value. The attribute value is a variable length field, which our system utilizes to publish encoded user information. To our knowledge, this is the first work that utilizes Bluetooth SDP by customizing service attributes to exchange non-service-related information.

However, SDP can only publish limited information, the size of which varies depending on brands and models of mobile phones. For example, in our experiments, one phone can publish up to 10 attributes, each of which has a maximum of 128 bytes of data. We use Bloom filters to encode and "compress" user information in order to fit it into SDP's attribute values. To further reduce the size of exchanged information, we propose a novel Bloom filter technique that iteratively refines Bloom filters in several rounds to achieve a desired low false positive rate given SDP's constraint. The Bloom filters are published via SDP to discover common topics. A device determines common topics by testing if its topics are in another device's Bloom filter. As a result, the system incurs limited transmission and computation. As a one-way hashing technique, Bloom filters also provide a reasonable level of privacy against eavesdroppers. It is

generally difficult, if not impossible, to reconstruct the information in a Bloom filter without performing an exhaustive search of the input space. [2]

Obviously, the interactions among autonomous and self interested entities can be modeled and analyzed as a socioeconomic system, and how to stimulate cooperative behaviors in such a system is an extensively studied topic in sociology and economics with a rich collection of analyzing techniques and promising solutions. Therefore, it is not surprising that all proposed incentive mechanisms for MWNs in the literature draw analogies from their counterparts in human societies.

The first category is barter based approaches, which are based on direct reciprocity: node A would provide resources/ services for node B only if B simultaneously provides resources/services for node A. This kind of bilateral and synchronous resource/service trading makes barter extremely simple to implement. From a system perspective, there is no need to keep any long-term state information, in the form of either reputation or currency, and as a consequence the implementation cost of barter is almost zero. However, synchronous trading is easy to fail when an action and its reward are not simultaneous. The second category is virtual-currency based, in which participating nodes would earn virtual currency by providing resources/services to others and spend the virtual currency to obtain resources/services from others. By taking virtual currency as a medium of exchange, nodes can then trade resources/services asynchronously, which overcomes the shortcoming of barters. Virtual currency, however, incurs a high implementation overhead, e.g., billing and e-cash transfers, implementations of centralized bank and dispute-resolution mechanisms, etc. In the third category, i.e., reputation based approaches, participants build up their reputation scores by providing services for others, and highly reputed participants receive preferential treatment when they need help. Obviously, reputation scores here can be treated as another form of virtual currency. Therefore, reputation based approaches share the same advantages and disadvantages as virtual-currency based ones. To sum up, existing incentive mechanisms are either less effective or incur high implementation costs, and therefore do not fit well with the requirements of MWNs. A new design paradigm is needed. [4]

Generally speaking, Private Set Intersection (PSI) is a cryptographic protocol that involves two players, Alice and Bob, each with a private set. Their goal is to compute the intersection of their respective sets, such that minimal information is revealed in the process. In other words, Alice and Bob should learn the elements (if any) common to both sets and nothing (or as little as possible) else. This can be a mutual process where, ideally, neither party has any advantage over the other.

Since mutual PSI can be easily obtained by two instantiations of one-way PSI (assuming that neither player aborts the protocol prematurely), in the remainder of this paper we focus on the latter. Hereafter, the term PSI denotes the one-way version and, instead of proverbial Alice and Bob, we use the terms client (C, i.e., the entity that learn the intersection) and server (S) to refer to the protocol participants.

One natural PSI extension is what we call PSI with Data Transfer or PSI-DT. In it, one or both parties have data associated with each element in the set, e.g., a database record. Data associated with each element in the intersection must be transferred to the client. It is also easy to see that PSI-DT is quite appealing in terms of actual database (rather than plain set) applications.

Another twist on PSI is the Authorized Version (APSI) where each element in the client set must be authorized (signed) by some recognized and mutually trusted authority. This requirement could be applicable to Examples 2 and 4. In the former, one or both agencies might want to make sure that names of terrorist suspects held by its counterpart are duly authorized by the country's top judiciary. In example 4, the bank could demand that each suspected tax cheat be pre-vetted by some international body, e.g., Interpol. In general, the main difference between PSI and APSI is that, in the former, the inputs (set items) of one or both parties might be arbitrarily chosen, i.e., frivolous. Clearly, other more interesting or more exotic variations are possible, e.g., the notion of group PSI with its many types of possible outputs. However, we limit the scope of this paper to the PSI flavors described above. [5]

III. PROPOSED SYSTEM

Our system consists of N users (parties) denoted as $P1, ..., PN$, each possessing a portable device. We denote the initiating party (*initiator*) as $P1$. $P1$ launches the matching process and its goal is to find one party that best *"matches"* with it, from the rest of the parties $P2, ..., PN$ which are called *candidates*. Each party Pi's profile consists of a set of attributes Si, which can be strings up to a certain length. $P1$ defines a matching query to be a subset of $S1$, and in the following we use $S1$ to denote the query set unless specified. Also, we denote $n = |S1|$ and $m = |Si|$, $i > 1$, assuming each candidate has the same set size for simplicity. Note that, we assume that the system adopts some standard way to describe every attribute, so that two attributes are exactly the same if they are the same semantically.

There could be various definitions of "match". In this paper, we consider a popular similarity criterion, namely the intersection set size $|S1 \cap Si|$. The larger the intersection set size, the higher the similarity between two users' profiles. User $P1$ can first find out her similarity with each other users via our protocols, and then will decide whether to connect with a best matching user based on their actual common attributes.

We assume devices communicate through wireless interfaces such as bluetooth or WIFI. For simplicity, we assume every participating device is in the communication range with each other. In addition, we assume that a secure communication channel has been established between each pair of users.

We *do not* assume the existence of a trusted third party during the protocol run; all parties carry out profile matching in a completely distributed way. They may cooperate with each other, i.e., when $P1$ runs the protocol with each Pi, a subset of the rest of parties would help them to compute their results. Note that, providing incentives for the users to cooperate is an important topic, and there are some existing mechanisms.

In this paper, we are mainly interested in insiders who are legitimate participators of the matching protocol and try to perform *user profiling*, i.e., obtain as much personal profile information of other nearby users as possible. For example, With a user's attributes, a bad guy could correlate and identify that user via its MAC addresses or public keys. However, we cannot absolutely prevent user profiling, because at least the initiator and its best matching user will mutually learn their intersection set. Thus we focus on minimizing the amount of private information revealed in one protocol run.

The main adversary model considered in this paper is *honest-but-curious* (HBC), i.e., a participant will infer private information from protocol run but honestly follow the protocol. Although we do not specifically address the malicious attacker model where an adversary may arbitrarily deviate from the protocol run, we will discuss how our protocols can be extended to achieve security in that model. The adversary may act alone or several parties may collude. We assume that the size of a coalition is smaller than a threshold t, where t is a parameter.

1) Security Goals: Since the users may have different privacy requirements and it takes different amount of efforts to achieve them, we hereby (informally) define two levels of privacy where the higher level leaks less information to the adversaries.

Privacy level 1 (PL-1): When the protocol ends, $P1$ and each candidate Pi, $2 \le i \le N$ mutually learn the intersection set between them: $I1,i = S1 \cap Si$. An adversary A should learn nothing beyond what can be derived from the above outputs and its private inputs. If we assume the adversary has unbounded computing power, PL-1 actually corresponds to unconditional security for all the parties under the HBC model. In PL-1, $P1$ can obtain all candidates' intersection sets within one protocol run, which may still reveal much user information to the attacker.

Privacy level 2 (PL-2): When the protocol ends, $P1$ and each candidate Pi, $2 \le i \le N$ mutually learn the size of their intersection set: $m1,i = |S1 \cap Si|$. The adversary A should learn nothing beyond what can be derived from the above outputs and its private inputs. In PL-2, except when $m1,i = |S1|$ or $|Si|$, $P1$ and each Pi both will not learn exactly which attributes are in $I1,i$. The adversary needs to run the protocol multiple times to obtain the same amount of information with what he can obtain under PL-1 when he assumes the role of $P1$.

2) Usability and Efficiency: For profile matching in MSN, it is desirable to involve as few human interactions as possible. In this paper, a human user only needs to explicitly participate in the end of the protocol run, e.g., decide whom to connect to based on the common interests. In addition, the system design should be *lightweight and practical*, i.e., being enough efficient in computation and communication to be used in MSN. Finally, different users (especially the candidates) shall have the option to flexibly *personalize their privacy levels*.

Design Challenges and Related Works

It is very challenging to achieve all the design goals simultaneously, especially if we desire high level of security but are unwilling to pay the cost of high computation and communication overhead. Similar problems to ours can be found in the literature, namely two-party private set intersection (PSI), private cardinality of set intersection (PCSI). Our privacy goals can be achieved if given multiple instances of PSI and PCSI, respectively. They are usually tackled with Secure Multi-party Computation (SMC). The general SMC techniques heavily rely on cryptography, and are well-known for their inefficiency. Researchers have proposed various customized solutions for those problems; for example, based on oblivious polynomial evaluation and oblivious pseudo-random functions that are secure in the HBC model. But when applied to the problem presented here, they incur either high computation or communication cost, thus are impractical to be used in MSN.

Concurrently with our work, a secure friend discovery protocol has been proposed. Different from us, their matching is based on computing the similarity (dot product) between two users' coordinates (which is not as intuitive as the intersection of the profile attributes as ours). In addition, a centralized trusted authority is needed to provide the coordinates. A private contact discovery protocol is proposed, where contact list manipulation is prevented by distributed certification. However, for general sensitive profile attributes it is difficult to find a distributed certifier in practice, whereas our protocols are not limited in the type of attributes to share with. Privacy-preserving multi-party interest sharing protocols for smartphone applications are proposed. However, their protocols rely on an online semi-trusted server, which may not be available when the users do not have connections to it.

In this paper, we propose two fully-distributed privacy preserving profile matching protocols, without relying on a client-server relationship nor any central server. We propose novel methods to reduce energy consumption and protocol run time, while achieving reasonable security levels. Specifically, we exploit the homomorphic properties of Shamir secret sharing to compute the intersection between user profiles privately, and due to the smaller computational domain of secret sharing, our protocols achieve higher performance and lower energy consumption for practical parameter settings of an MSN. Such a framework is also applicable to many scenarios beyond the motivating problems in this paper, for example, in patient matching in online healthcare social networks.

IV. RESULTS

The concept of this paper is implemented and different results are shown below, The proposed paper is implemented

in NS 2.34 on fedora on a Pentium-IV PC with minimum 20 GB hard-disk and 1GB RAM. The propose paper's concepts shows efficient results and has been efficiently tested on different scenarios.

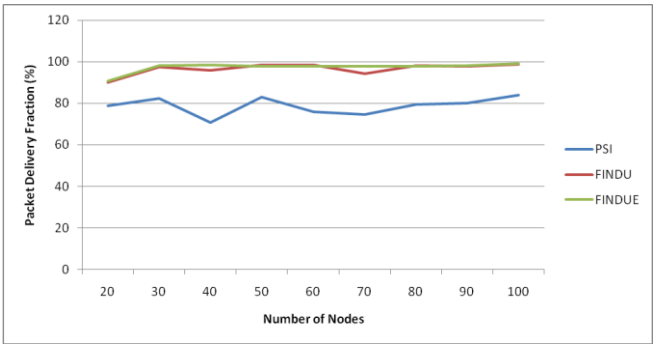

Figure 2: Number of Nodes Vs Packet Delivery Fraction

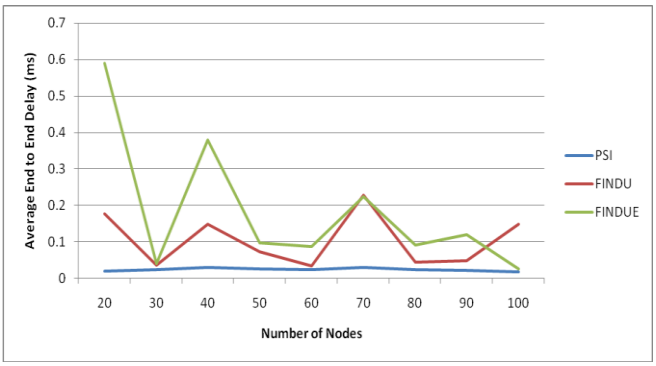

Figure 3: Number of Nodes Vs Average End to End Delay

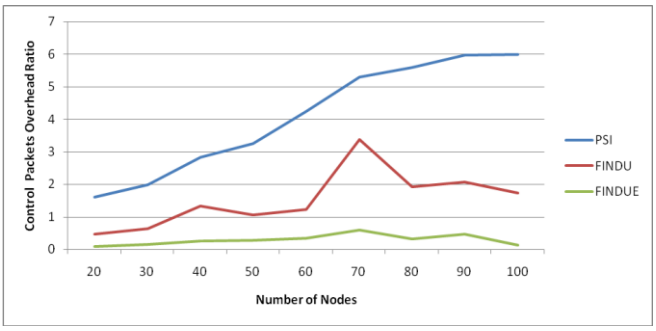

Figure 4: Number of Nodes Vs Control Overhead

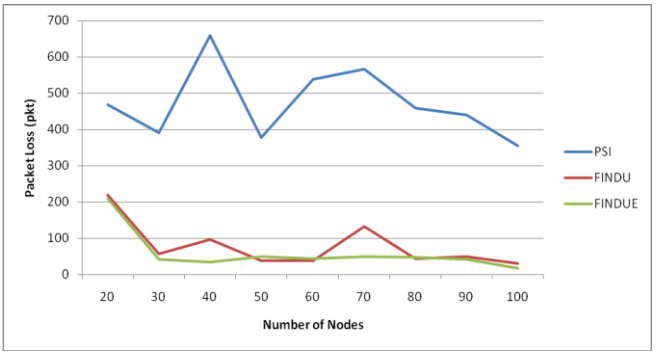

Figure 5: Number of Nodes Vs Packet Loss

V. CONCLUSIONS

In this paper, we for the first time formalize the problem of privacy-preserving distributed profile matching in MSNs, and propose two concrete schemes that achieve increasing levels of user privacy preservation. Towards designing lightweight protocols, we utilize Shamir secret sharing as the main secure computation technique, while we propose additional enhancements to lower the proposed schemes' communication costs. Through extensive security analysis and simulation study, we show that 1) our schemes are proven secure under the HBC model, and can be easily extended to prevent certain active attacks; 2) our schemes are much more efficient than state-of the art ones in MSNs where the network size is in the order of tens, and when the number of query attributes is smaller than the number of profile attributes.

REFERENCES

[1] M. Li, N. Cao, S. Yu, and W. Lou, "Findu: privacy-preserving personal profile matching in mobile social networks," in *Proc. 2011 IEEE INFOCOM*, pp. 1–9.
[2] Z. Yang, B. Zhang, J. Dai, A. Champion, D. Xuan, and D. Li, "E-smalltalker: a distributed mobile system for social networking in physical proximity," in *2010 IEEE ICDCS*.
[3] R. Lu, X. Lin, X. Liang, and X. Shen, "A secure handshake scheme with symptoms-matching for mhealthcare social network," *2010 Mobile Netw. Applications*, pp. 1–12.
[4] C. Zhang, X. Zhu, Y. Song, and Y. Fang, "C4: a new paradigm for providing incentives in multi-hop wireless networks," in *Proc. 2011 IEEE INFOCOM*, pp. 918–926.
[5] E. D. Cristofaro and G. Tsudik, "Practical private set intersection protocols with linear complexity," in *2010 Financial Cryptography and Data Security*.
[6] W. Dong, V. Dave, L. Qiu, and Y. Zhang, "Secure friend discovery in mobile social networks," in *Proc. 2011 IEEE INFOCOM*, pp. 1–9.
[7] E. De Cristofaro, M. Manulis, and B. Poettering, "Private discovery of common social contacts," in *Proc. 2011 Applied Cryptography and Network Security*, pp. 147–165.
[8] E. De Cristofaro, A. Durussel, and I. Aad, "Reclaiming privacy for smartphone applications," in *Proc. 2011 IEEE International Conf. Pervasive Comput. Commun.*, pp. 84–92.
[9] M. Li, S. Yu, J. D. Guttman, W. Lou, and K. Ren, "Secure ad-hoc trust initialization and key management in wireless body area networks," *ACM Trans. Sensor Netw.*, 2012.
[10] S. Gollakota, N. Ahmed, N. Zeldovich, and D. Katabi, "Secure in-band wireless pairing," in *Proc. 2011 USENIX Conf. Security*, pp. 16–16.
[11] M. Li, S. Yu, N. Cao, and W. Lou, "Privacy-preserving distributed profile matching in proximity-based mobile social networks," technical Report, 2012. Available: http://digital.cs.usu.edu/ ~mingli/papers/Findu techrep.pdf.
[12] G. T. E. De Cristofaro and J. Kim, "Linear-complexity private set intersection protocols secure in malicious model." in *2010 Asiacrypt.*

Author:-

Sai Sudha persuing M.Tech in the field of Digital systems and computer electronics from Jawaharlal Nehru Technological University, Hyderabad, India.

T.Madhavi Kumari Associate Professor Electronics & Communication Engineering **department and Co-Ordinator ,Academic & Planing of** Jawaharlal Nehru Technological University, Hyderabad, India.