ISSN 2047-3338

Multi-Factor User Authentication in Wireless Sensor Networks

T. Sarika[1] and Shaik Shah Nawaz[2]
[1,2]Aurora's Engineering College, Bhongir, A.P., India

***Abstract*– Wireless Sensor networks are a new wireless networking paradigm for the mobile hosts. Wireless Sensor networks are an autonomous system that is made up of collaborative mobile nodes. Wireless Sensor networks can be dynamically setup without relying on any pre-existing infrastructure. Implementing the public key user authentication is a challenging issue in Wireless Sensor networks due to its salient nature of the network. The user authentication is of three steps. In the first step Registration of user will be happened, in the second step user smart card will be validated with symmetric key and using the key the authentication of the user will be done as a third phase.**

***Index Terms*– Authentication, Sensor Networks, Security and Hash Function**

I. INTRODUCTION

OVER the years, Wireless Sensor Networks (WSN) have attracted in increasing interest from researchers due to its ubiquitous nature, easy deployment, and the range of applications they enable. Networks of thousands tiny sensor devices, which have low processing power, limited memory and energy [1], play important roles for an economical solution to some of the challenging problems, such as, real-time traffic monitoring, building safety monitoring, military sensing and tracking, wildlife monitoring, measurement of seismic activity and so on.

In general, most of the queries in WSN applications are issued at the points of base stations or Gateway (GW) nodes of the network. However, one can foresee that there should have great needs to access the real-time data inside WSN, where real-time data from the sensor nodes may no longer be accessed through the GW-node only, instead, the data are to be accessed directly by the external party (user) as and when demanded. If the data in WSN are made available to the user on demand, then authentication of the user must be if the data in WSN are made available to the user on demand, then authentication of the user must be ensured before allowing the user to access been addressed adequately in comparisons with the network and link layers protocols [2], [3], [4] in WSN. One of the possible factors could be the resource constrained WSN environment. In this letter, we present an efficient user authentication protocol for WSN. The data encryption algorithm is divided into symmetrical encryption algorithm and asymmetrical encryption algorithm, the commonly used symmetrical encryption algorithm has AES, DES, and 3DES and so on, the commonly used asymmetrical encryption algorithm has ECC, RSA and so on. In this paper, the improved AES algorithm is used to encrypt plaintext. Thus we have solved the question of the operating speed and the key allocation management. The protocol uses the multi-factor authentication concept and resists many logged in users with the same login identity, stolen-verifier, guessing, and impersonation and replay threats. A multi factor authentication is a concept used to describe an authentication mechanism, where more than one factor (e.g., password, smart card and OTP) is required to authenticate the communicating party. We refer readers to [5], [6] for more details on multi-factor authentication mechanism. The remainder of the paper is organized as follows. Section 2&3, reviews the related work and discusses the security requirements in WSN. Section 4 presents our proposed Framework and Section 5 analyzes the Frame work, Section 6 concludes the letter.

II. RELATED WORK

AES algorithm: AES is the norms of electronic data encryption used by the U.S. National Institute of Standards and Technology. It is a symmetric block cipher system. It uses replace / exchange network. The data block length and key length are variable. Three key lengths: 128,192, 256, whose iteration cycle Nr is 10, 12 and14 round respectively, are used. The AES algorithm mainly includes three aspects: round change, turns and key expand. Each round transformation is composed by the non-linear layer, the linear mixture layer, the add roundkey layer.

Security Requirements in WSN: Sastry and Wagner observed the merits and limitations of security aspects of the IEEE 802.15.4 specification. The specification allows a maximum of 255 Access Control List (ACL) entries, where within ACL there is no support for group keying and pair wise keying in IEEE 802.15.4 specification. The specification suffers from IV Management, Key management problems and insufficient integrity protection. It should be noted that the IEEE 802.15.4 API indicates two clear directions. One of them is to go with the specification itself without adding more security patches, and the other is to adopt add-on security service on top of the API according to application's requirement. Nevertheless, one has to see that the combined

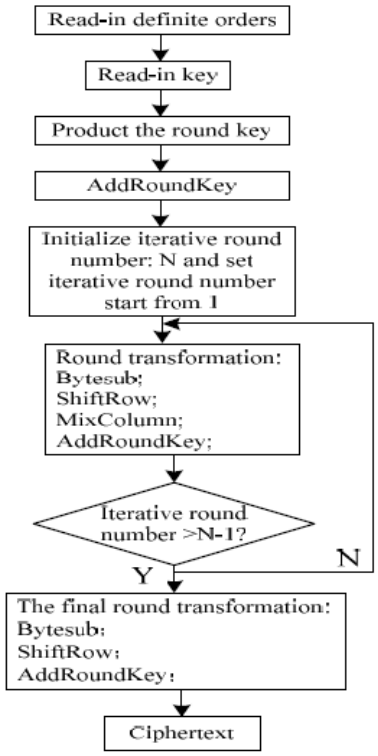

Fig 1: AES encryption process scheme

security suite must consider some basic security requirements, AES encryption process is shown in Fig 1:

A. Realization of AES Algorithm

AES is iterative block encryption algorithm that has the variable length data block and the variable length key. After several rounds of block transform, an intermediate state of transformation is generated. The state can be expressed as a two-dimensional byte array, which has 4 line and Nb column (Nb is the data block length divided by 32). Keys can also be expressed as a two-dimensional byte array, which has 4 line and Nk column (Nk is the key block length divided by 32). The transform round number Nr is decided by Nb and Nk, as shown in Table 1.

Each round transformation contains 4 steps, which are bytesub, shift row, mix column, addroundkey respectively:

(1) Bytesub is non-linear permutation based on Sbox. It is used to map each byte of input or intermediate state into another byte through a simple table look-up operation. Then elements of rows and columns of S-box are taken out as output.

(2) ShiftRow is a line-based operator of cyclic shift. The last 3 rows of ShiftRow state is followed by cyclic shift in turn. For example, when Nb=4, the 0th line is cyclic left 0 bit, the 1st line is cyclic left 1 bit, the 2nd line is cyclic left 2 bit, the 3rd line is cyclic left 3 bit.

TABLE 1: THE RELATIONSHIP BETWEEN THE TRANSFORM ROUND NUMBER Nr AND Nb, Nk

Nr	Nb=4	Nb=6	Nb=8
Nk=4	10	12	14
Nk=6	12	12	14
Nk=8	14	14	14

(3) Mixcolumn is a replacement operation. It runs plus and multiplication operation of the value of state-byte in mathematical domain. So each byte is replaced by the result.

(4) The state adjustment result is obtained by XOR of the addroundkey with a roundkey:

$$(a_{i,j})_{4\times4} \otimes (K_{i,j})_{4\times4} = (b_{i,j})_{4\times4} \quad (1)$$

Where ai,j is each byte of state data block Nb=4, ki,j is each byte of state round key Nk=4, bi,j is the result of XOR. Round key is derived from the key, including key Expansion and round key. The realization of the process is as follows [3]:

(1) The key is expanded to inflation key.

(2) The sum of bit number of round key is equal to the value that is the block length multiplied by a round number added 1. For example, when Nb=4, Nr=12, the sum of bit number is 128× (12+l) =1664 bit.

(3) The round key is taken out from inflation key. The 1st round key is composed by the first Nb characters, the 2nd round key by next Nb characters and so on.

B. An Improved AES Algorithm

S-box is an independent non-linear transform for state bytes. The design criteria are used to achieve the confusion function of Shannon password, so its nature often decides the security of the entire password.

The AES S-box transform is composed by the inverse of multiplication and the affine transformation.

(1) The inverse of multiplication is defined as following. When input is $x\in GF(2^8)$, we need to get $X=(x)^{-1}$, Where $(x)^{-1}$ is as follows:

$$x=(x)^{-1}=_{i}(x)^{254}, x\neq0;\ 0,\ x=0; \quad (2)$$

Where GF (2^8) represents the finite field of 8-th power of the prime numbers 2. The order of this finite field is 256.

(2) The affine transformation is defined as following [4]: For$\forall a\in GF(2^8)$ its affine transformation is:

$$S_{u,v}(a)=b=(b_7b_6\ldots b_0) \quad (3)$$

Where $(b_7b_6\ldots b_0)$ is expressed by:

$b(x)= b_0 + b_1x +\ldots. b_7 x^7$,and $(a_7a_6\ldots a_0)$ by $a(x)= a_0 + a_1x +\ldots. a_7 x^7$,

Eq. (3) can be written as:

$$b(x)=u(x)+v(x)\bmod(x^8+1) \quad (4)$$

Matrix form of Eq. (4) is as follows:

$$\begin{bmatrix} b_0\\ b_1\\ b_2\\ b_3\\ b_4\\ b_5\\ b_6\\ b_7 \end{bmatrix} = \begin{bmatrix} 10001111\\ 11000111\\ 11100011\\ 11110001\\ 11111000\\ 01111100\\ 00111110\\ 00011111 \end{bmatrix} \begin{bmatrix} a_0\\ a_1\\ a_2\\ a_3\\ a_4\\ a_5\\ a_6\\ a_7 \end{bmatrix} + \begin{bmatrix} 1\\ 1\\ 0\\ 0\\ 0\\ 1\\ 1\\ 0 \end{bmatrix} \quad (5)$$

which $u(x)=x^7+x^6+x^5+x^4+1$, $v(x)=x^7+x^6+x^2+x$.

S-box produced by above method exists some shortcomings, such as its iteration cycle is excessively short (not bigger than 88), periodicity is poor, the powerful difference attack is caused [5]. Eventually the password is deciphered. In addition S box algebraic expression is as follows:

$$Y=05x^{254}+09\ x^{253}+f9\ x^{251}+25\ x^{247}+f4\ x^{239}+01\quad x^{223}+b5\ x^{191}+8fx^{127}+63 \qquad (6)$$

Which is too simple (only 9 items). So S-box cannot guarantee the security, resisted from the AES algebra computation attacks [6].For short algebraic expression of S-box, literature [7] proposes a method that exchanges the order of inverse of multiplication and affine transformation computation to solve this problem.

This method can increase the length of algebraic expression from 9 items to 255 items. The literature [8] proposes new affine transform pair based on the S-box, which can make the S-box's iteration cycle to traversal entire 256 spaces of GF (2^8).

In this paper, we introduce a new S-box alternative scheme, which enables the iteration cycle of each element in GF (2^8) to achieve 256. That is, all elements of GF (2^8) only belong to one iterative cycle. The new affine transform pair (155, 62) will possibly expand the S-box iteration cycle. In addition we firstly have the affine transformation, next have the inverse of multiplication, which can make the S-box algebraic expression very complex. So AES have strongly ability to resist algebra computation attack. The improved affine transformation is:

$$Su,v(ai,j) = u(x)\ aij(x)+v(x)\ mod(x^8+1) \qquad (7)$$

Where $u(x) = x^7+ x^6+ x^4+ x^3+ 1$,

$v(x) = x^5+ x^4+ x^3+ x^2+ x$, we get u=155,v=62;

The procedure of improved S-box operation is as follows:

(1) Select the new affine transform pair (155, 62) to make the following operation, as (8):

$$a'(x) = ua(x) + v = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \begin{bmatrix} 11011010 \\ 01101101 \\ 10110110 \\ 01011011 \\ 10101101 \\ 11010110 \\ 01101011 \\ 10110101 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \qquad (8)$$

(2) Run the inverse of multiplication, such as

$$a^{n}(x)=a^{-1}(x)=\{a'(x)^{256},\ a'(x)\neq 0 \qquad (9)$$

$$0,\ a'(x)=0$$

III. PROPOSED FRAMEWORK

In this section, we propose a user authentication framework for wireless sensor networks that ensures the access and supply of data are taking place by the legitimate users only. So, in this regards, a user must register with the gateway node in a secure manner to access the real time sensors data. Upon the successful user registration, the gateway node personalizes a smart card to every registered user. Then, a user can submit his/her query in an authentic way and access the sensor networks data at any time within an administrative configurable period. The framework is divided into four phases, namely, user registration phase, login phase, authentication phase and password change phase. For convenience Table 2, a list of some notations and symbols will be used throughout the rest of paper.

TABLE 2: NOTATION AND SYMBOL USED IN PAPER

Notation	Descriptions
U_k	User k^{th} to be login
ID_k	Login_ID of U_k
PW_k	Password of U_k
FID_k	Fictitious ID of U_k
GW node	WSN gateway node
Sn	Sensor Node
X_g	Secret parameter generated by GW node
h (·)	Cryptographic hash function
⊕	Bitwise XOR function
||	Bitwise concatenation function

We have made the following assumptions for multi-factor user authentication framework:

- All the nodes locations are static.
- The GW node is considered as trusted node.
- Each user who wishes to access the sensor network data, he/she must register once with the GW node.

The secret parameter X_g is not known to the user,

Since: it is generated and securely stored in user's smart card by the GW node.

A. Registration Phase (RP)

In the registration phase, initially, each user must register him/her with the GW node using following steps:

RP-1). The user U_k submits his/her identity and password (i.e., ID_k and PW_k) to the GW node over a secure channel.

RP-2). Upon receiving the U_k registration request, the GW

- node computes:$h(PW_k)$
- $h(X_g)$
- $Nk= h(ID_k||h(PW_k)||h(X_g))$

Thereafter, the GW node personalizes a smart card to the user with the parameters {h (.), ID_k, Nk, $h(PW_k)$, $h(X_g)$}, here, X_g is secret parameter securely generated and hashed by the GW node. Furthermore, $h(X_g)$ is stored in the sensor nodes prior to deployment of the sensor nodes, who wish to exchange their data with the users. This step completes the user's registration phase.

B. Login Phase (LP)

This phase invoked whenever a user U_k wants to submit his/her query to access the sensor networks, then every time he/she has to accomplish the login phase. The user U_k inputs his ID_k and PW_k and inserts his smart card in the terminal.

Now, smart card confirms the entered *IDk* and PW_k with thepre-stored values (ID_k and PW_k *)*. If the entered ID_k and PW_k are not verified then reject the login request. Otherwise, smart card executes some functions and performs following:

LP-1).Compute $FID_k=h(ID_k \| h(PW_k) \| h(X_g)) \oplus h(h(X_g) \| T)$, here, T is denoting the current timestamp of U_k's system.

LP-2). Compute $C_k= h(N_k \| h(X_g) \| T)$

LP-3). Now, this is end of the login phase and user login request message <FID_k, C_k, T> is send to the GW node over a secure channel for the authentication process.

C. Authentication Phase (AP)

The authentication phase is invoked when it receives the User's login request message <FID_k , C_k, T> at time T*'*. Now, the GW node authenticates user requests by the following steps:

AP-1). The GW node*s* validate the T: check, if $(T^{'}-T) \geq \Delta T$, if yes, then the GW node rejects this request and terminate the process. Otherwise, continues with next step. Here, T*'* is the current timestamp of the GW node and ΔT is the expected time interval for the transmission delay.

AP-2). Computes:

$h(ID_k||h(PW_k)||h(X_g))=FID_k \oplus h(h(X_g)||T)$, and $C_k=h(h(ID_k||h(PW_k)||h(X_g))||h(X_g)||T)$.

AP-3). Thereafter check: if (C_k*==C_k) holds then the GW node accepts the login request, and proceeds to next step, otherwise request rejects and terminates the further operations.

AP-4). And the GW node computes:

$A_k=h(FID_k||S_n||h(X_g)||T'')$.

AP-5). Now, the GW node sends a user's request message <FID_k , A_k, T''> over a communication channel to some nearest sensor node (e.g., S_n), which user was demanding. Here, message $A_k=h(FID_k||S_n||h(X_g)||T'')$. and T''is the current timestamps of the *GW node*. Furthermore, *Ak* make sure to the sensor node that the request message <FID_k , A_k, T''> has comes from the authentic GW node and not from the fake. Since, A_k is computed with help of the GW node secret parameter (h (X_g)), which is known to the GW node and securely stored in the sensor nodes.

AP-6). Now, S_n validate the time T''*:* check, if $(T'''-T'') \geq \Delta T$ if yes, then the sensor node (S_n) rejects the request and terminate the process, otherwise continues with next step. Here, *T'''* is the current timestamp of the sensor node and ΔT is the expected time interval for the transmission delay.

AP-7). S_n now computes $A_k=h(FID_k||S_n||h(X_g)||T'')$. and check: if ($A_k$==$A_k$) holds then the sensor node respond to user's requested query. Otherwise rejects the request and terminates the further operations.

AP-8). S_n now computes message $M_k=h(S_n||h(X_g)||T''')$ and provide mutual authentication to the GW node. Here, T'''' is the current timestamp of *Sn* node's and sends message <M_k,T'''' > to the GW node.

AP-9). Upon receiving the message < M_k, T''' >, the GW node validate the time T'''*:* check, if $(T'''-T'') \geq \Delta T$ if yes, then the GW node rejects the request and terminate the process, otherwise continues with next step. Here, *T** is the current timestamp of GW node and ΔT is the expected time interval for the transmission delay.

AP-10) The GW node now computes M_k *= $h(S_n||h(X_g)||T''')$ and check: if (M_k* == M_k) holds then mutual authentication is take places between the sensor node *Sn* and the GW node. Otherwise, S_n is not an authentic node and the GW node terminates the operations.

AP-11). Upon the successful mutual authentication between the sensor node S_n and the GW node, user U_k starts the access of sensor data. The sensor node S_n and the GW node, user U_k starts the access of sensor data.

D. Password-Change Phase (PP)

The password change phase is invoked whenever user U_k wants to change or update his/her old password (PW_k*)* to a new password, say PW_k*. The password change phase is described in the following:

PP-1). User U_k inserts his/her smart card into the terminal and enters his/her identity (ID_k) and password (PW_k).

PP-2). Now the smart card validates the U_k's entered ID_k and PW_k with stored values and check: if holds then the smart card request to the user for new password (i.e., PW_k*). Otherwise, rejects the password change request and terminates the operation.

PP-3). Now upon receiving the U_k new password (PW_k*), smartcard computes:

N_k*= $N_k \oplus h(ID_k||h(PW_k)||h(X_g)) \oplus h(ID_k||h(PW_k*)||h(X_g))$, here N_k is old stored value of the smart card (i.e., N_k= $h(ID_k||h(PW_k)||h(X_g))$).

PP-4). Now the smart card replaces the old values (N_k and $h(PW_k)$) with new values(N_k*and $h(PW_k*)$).After performing the above steps, password change phase successfully takes places.

IV. ANALYSIS OF FRAMEWORK

In this section, we present our proposed multi-factor framework strength in terms of security analysis (i.e., resist against several well-known attacks) and performance analysis.

A. Security Analysis

Now we will shows the proposed framework that resists against the following attacks: such as, replay attack, impersonation attack, stolen-verifier attack, password guessing attack, node-compromise attack, man-in-the middle attack, denial-of-service attack. Furthermore, our framework consider mutual authentication between the GW node and the sensor node and enable users to change their password freely whenever required.

1). Replay attack*:* Our two-factor framework is secure against a replay attack, because the authenticity of messages <FID_k,C_k,T>, <FID_k,A_k,T''> and <M_k,T''' > is validated by checking the freshness of three timestamps $(T'-T) \geq \Delta T$,$(T'''-T'') \geq \Delta T$ and $(T*-T''') \geq \Delta T$ Thus, our framework is secure against replaying of message.

2). Impersonation attack*:* An attacker cannot impersonate the user, suppose, an attacker captures a login message <FID_k,C_k,T>. Now, he/she will wish to login again into the system, but this is not feasible to get the original password (PW_k) and secret (X_g), since, they are hashed with the cryptographic hash functions. Therefore, it is not possible to impersonate the user.

3). Stolen-verifier attack: Stolen-verifier attack is not applicable to our framework, because we are not using any password/verifier table. So, any kind of stolen verifier attack will not occur on our framework.

4). Password guessing attack: Since, our framework is free from any password verifier table, so password guessing attack is not feasible. Furthermore, in login phase the password is not simply transmitted but it is transmitted with some other secrets, which is difficult to guess the user password.

5). Node-compromise attack*:* In general, sensors are placed in the hostile environment to do their task. Therefore, it could easy to capture a node from the environment and steals stored secret information about the networks. In order to overcome this attack, we store hashed secret values in sensor node, as $A_k=h(FID_k||S_n||h(X_g)||T")$.Thus, by doing so, it is difficult to get the secret parameters in the case of node compromised.

6). Man-in-the-middle attack: An attacker may attempt a man-in- the-middle (MITM) by modifying the login message $<FID_k,C_k,T>$.into $<FID_k*,C_k*,T*>$.However, this malicious attempt will not works, because without the knowledge of Xg, it is not easy to calculate the message $<FID_k*,C_k*,T*>$.Thus, man-in-the-middle attack is not applicable in our framework.

7). Denial-of-service attack*:* In password change phase, denial of-service attack cannot work in our framework. Suppose, when user smart card is stolen, unauthorized users cannot change new password. Since, in password change phase the old ID_k and PW_k is verified first. Hence, denial-of-service attack is not applicable on stolen smart.

8). Secure password change*:* In secure password change or update phase, our framework first verifies the old password, identity and then only request for new password. Otherwise rejects the password change requests. Therefore, our framework change password securely.

B. Performance Analysis

Here we examine the performance and summarize the security functionality of our multi-factor framework and compare with the M.L. Das multi-factor scheme. Table 3 shows that our framework is more secure and robust as compares to Das's scheme and provides more security features on reasonable computational costs. As shown in Table 3, our two-factor framework has 13H computational cost and provides robust security features, such as, secure password change phase, provide mutual authentication, secure against node compromised attack and secure against denial-of-service attack and secure against many other attacks.

V. CONCLUSION

In this paper, we have analyzed the security weaknesses in a two-factor user authentication protocol for wireless sensor networks. We have provided security evaluation and efficiency analysis, which show that our protocol is more robust and secure than the existing schemes and as efficient as them.

TABLE 3: PERFORMANCE ANALYSIS

Security Features	Our Scheme	Das's [1]
Secure password change phase	Yes	No
Provides mutual authentication	Yes	No
Secure against node compromised attack	Yes	No
Secure against denial-of-service	Yes	No
Registration phase computational cost	3H	2H
Login phase computational cost	3H	3H
Authentication/verification phase cost	7H	5H

*H: A cryptographic hash function

REFERENCES

[1] Manik Lal Das."Two-Factor User Authentication in Wireless Sensor Networks", IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, 2008, 8(3):p.1 086-1090.

[2] Wang Zheng-cai, Yang Shi-ping."Research on Methods for Designing Authentication Protocols Attack". Computer Engineering Design, 2008, 29(20):p.5163-5165, 5170.

[3] Hsiang H. and Shih W. (2009). Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment. Computer Standards & Interfaces, doi: 10.1016j.csi.2008.11.002.

[4] KuW. Chen S. (2005). Cryptanalysis of a flexible remote user authentication scheme using smart cards. ACM Operating Systems Review, vol. 39, no. 1, pp. 90–96.

[5] Tsern, H.L. Simple Dynamic User Authentication Protocols for Wireless Sensor Networks, In Proceedings of 2nd International Conference on Sensor Technologies and Applications, Cap Esterel, France, 2008; pp. 657-660.

[6] Nyang, DH.; Lee M.K. Improvement of Das's Two-Factor Authentication Protocol in Wireless Sensor Networks, Cryptology ePrint Archive 2009/631. Online PDF: http://eprint.iacr.org/ 2009/631.pdf (accessed on 28 February 2010)

[7] J. Lee JH. Song and T. Iwata. The AES-CMAC algorithm, IETF Request for Comments RFC-4493, 2006.

[8] R. Watro, D. Kong, S. Cuti, C. Gardiner, C. Lynn, and P. Kruus, "TinyPK: securing sensor networks with public key technology," in Proc.ACM Workshop Security of Ad Hoc Sensor Networks, pp. 59-64, 2004.

Shaik Shah Nawaz has received his M. Tech degree in Computer Science. Currently, he is working as Senior Associate Professor in the Department of Computer Science & Engineering, Aurora Engineering College, Bhongir, Nalgonda, A.P, and India. He has 18-years working Experience in Software Industry. His areas of interests are Software Engineering, Testing, Network Security, Computer Networks, Wireless Communications, Data Mining and Data Warehousing.

T. Sarika is pursuing his M. Tech in Software Engineering (Dept. of CSE) in Aurora Engineering College, Bhongir, Nalgonda, A.P and India. His areas of interests are Software Engineering, Testing, Wireless Networks and Mobile Computing.