International Journal of Research Studies in Science, Engineering and Technology
Volume 1, Issue 8, November 2014, PP 1-8
ISSN 2349-4751 (Print) & ISSN 2349-476X (Online)

Discovery and Verification of Neighbor Positions in Mobile AD HOC Networks

Anil Kumar Gona[1], Ratna Raju Mukiri [2]

[1] PG Scholar, Eswar College of Engineering, Narasaraopet, Guntur.
[2] Asst. Professor, Eswar College of Engineering, Narasaraopet, Guntur.

Abstract: *A growing number of ad hoc networking protocols and location-aware services require that mobile nodes learn the position of their neighbors. However, such a process can be easily abused or disrupted by adversarial nodes. In absence of a priori trusted nodes, the discovery and verification of neighbor positions presents challenges that have been scarcely investigated in the literature. In this paper, we address this open issue by proposing a fully distributed cooperative solution that is robust against independent and colluding adversaries, and can be impaired only by an overwhelming presence of adversaries. Results show that our protocol canthwart more than 99 percent of the attacks under the best possible conditions for the adversaries, with minimal false positive rates.*

1. INTRODUCTION

Location awareness has become an asset in mobile systems, where a wide range of protocols and applications require knowledge of the position of the participating nodes. Geographic routing in spontaneous networks, data gathering in sensor networks, movement coordination among autonomous robotic nodes, location-specific services for handheld devices, and danger warning or traffic monitoring in vehicular networks are all examples of services that build on the availability of neighbor position information. The correctness of node locations is therefore an all important issue in mobile networks, and it becomes particularly challenging in the presence of adversaries aiming at harming the system. In these cases, we need solutions that let nodes 1) correctly establish their location in spite of attacks feeding false location information, and 2) verify the positions of their neighbors, so as to detect adversarial nodes announcing false locations. In this paper, we focus on the latter aspect, hereinafter referred to as neighbor position verification (NPV for short). Specifically, we deal with a mobile ad hoc network, where a pervasive infrastructure is not present, and the location data must be obtained through node-to-node communication by advertising forged positions, adversaries could bias geographic routing or data gathering processes, attracting network traffic and then eavesdropping or discarding it. Similarly, counterfeit positions could grant adversaries unauthorized access to location- dependent services, let vehicles forfeit road tolls, disrupt vehicular traffic or endanger passengers and drivers. In this context, the challenge is to perform, in absence of trusted nodes, a fully distributed, lightweight NPV procedure that enables each node to acquire the locations advertised by its neighbors, and assess their truthfulness. We therefore propose an NPV protocol hat has the following features. It is designed for spontaneous ad hoc environments, and, as such, it does not rely on the presence of a trusted infrastructure or of a priori trustworthy nodes; . It leverages cooperation but allows a node to perform all verification procedures autonomously. This approach has no need for lengthy interactions, e.g., to reach a consensus among multiple nodes, making our scheme suitable for both low- and high mobility environments; . It is reactive, meaning that it can be executed by any node, at any point in time, without prior knowledge of the neighborhood; . It is robust against independent and colluding adversaries, It is lightweight, as it generates low overhead traffic.

2. LITERATURE SURVEY

Although the literature carries a multitude of ad hoc security protocols addressing a number of problems related to NPV, there are no lightweight, robust solutions to NPV that can operate autonomously in an open, ephemeral environment, without relying on trusted nodes. Below, we list relevant works and highlight the novelty of our contribution. For clarity of presentation, we first

review solutions to some NPV-related problems, such as secure positioning and secure discovery, and then we discuss solutions specifically addressing NPV. Securely determining own location. In mobile environments, self-localization is mainly achieved through Global Navigation Satellite Systems, e.g., GPS, whose security can be provided by cryptographic and noncryptographic defense mechanisms. Alternatively, terrestrial special purpose infrastructure could be used along with techniques to deal with non honest beacons. We remark that this problem is orthogonal to the problem of NPV. In the rest of this paper, we will assume that devices employ one of the techniques above to securely determine their own position and time reference. Secure neighbor discovery (SND) deals with the identification of nodes with which a communication link can be established or that are within a given distance. SND is only a step toward the solution we are after: simply put, an adversarial node could be securely discovered as Neighbor and be indeed a neighbor (within some SND range), but it could still cheat about its position within the same range. In other words, SND is a subset of the NPV problem, since it lets node assess whether another node is an actual neighbor but it does not verify the location it claims to be at. SND is most often employed to counter wormhole attacks. Practical solutions to the SND problem have been proposed. While properties of SND protocols with proven secure solutions. Neighbor position verification was studied in the context of ad hoc and sensor networks; however, existing NPV schemes often rely on fixed. Trust worthy nodes, which are assumed to be always available for the verification of the positions announced by third parties. In ad hoc environments, however, the pervasive presence of either infrastructure or neighbor nodes that can be aprioristically trusted is quite unrealistic. Thus, we devise a protocol that is autonomous and does not require trustworthy neighbors. an NPV protocol is proposed that first lets nodes calculate distances to all neighbors, and then commends that all triplets of nodes encircling a pair of other nodes act as verifiers of the pair's positions. This scheme does not rely on trustworthy nodes, but it is designed for static sensor networks, and requires lengthy multiround computations involving several nodes that seek consensus on a common neighbor verification. Furthermore, the resilience of the protocol . to colluding attackers has not been demonstrated To our knowledge, our protocol is the first to provide a fully distributed, lightweight solution to the NPV problem that does not require any infrastructure or a priori trusted neighbors and is robust to several different attacks, including coordinated attacks by colluding adversaries. Also, unlike previous works, our solution is suitable for both low and high mobile environments and it only assumes RF communication. Indeed, non-RF communication, e.g., infrared or ultrasound, is unfeasible in mobile networks, where non-line-of-sight conditions are frequent and device to device distances can be in the order of tens or hundreds of meters. An early version of this work, sketching the NPV protocol and some of the verification tests to detect independent adversaries,

3. EXISTING SYSTEM

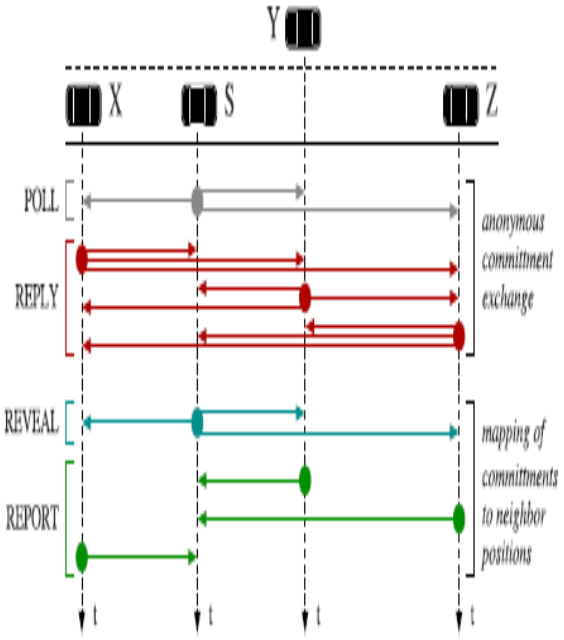


Fig. 1. *Message exchange overview, during one instance of the NPV protocol*

Geographic routing in spontaneous networks, data gathering in sensor networks, movement coordination among autonomous robotic nodes, location-specific services for handheld devices, and danger warning or traffic monitoring in vehicular networks are all examples of services that build on the availability of neighbor position information.

The correctness of node locations is therefore an all important issue in mobile networks, and it becomes particularly challenging in the presence of adversaries aiming at harming the system. We consider a mobile network and define as communication neighbors of a node all the other nodes that it can reach directly with its transmissions .We assume that each node knows its own position with some maximum error_p, and that it shares a common time reference with the other nodes: both requirements can be met by equipping communication nodes with GPS receivers.1 In addition; nodes can perform Time-of-Flight-based RF ranging with a maximum error equal to _r. As discussed in .this is a reasonable assumption, although it requires modifications to off-the-shelf radio interfaces; also, promising techniques for precise ToF-based RF ranging have been developed .

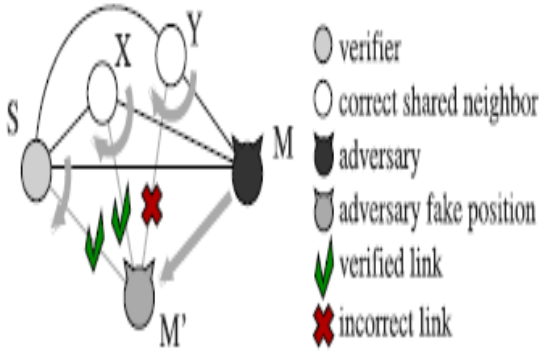


Fig. 2. *Example of topological information stored by verifier S at the end*

Of the message exchange and effect of a fake position announcement By M. We assume that node positions do not vary significantly during a protocol execution, since a complete message exchange takes no more than a few hundreds of milliseconds. The relative spatial movements of the nodes during such a period are taken into account through the tolerance value _m. Nodes carry a unique identity2 and can authenticate messages of other nodes through public key cryptography.In particular, we assume that each node X owns a private key, kX, and a public key, KX, as well as a set of one-time use keys {k0 X;K0X }, as proposed in emerging architectures for secure and privacy-enhancing communication . Node X can encrypt and decrypt data with its keys and the public keys of other nodes; also, it can produce digital signatures (SigX) with its private key. We assume that the binding between X and KX can be validated by any node, as in state-of-the-art secure communication architectures.

Disadvantages of Existing System

Correctly establish their location in spite of attacks feeding false location information, and Verify the positions of their neighbors, so as to detect adversarial nodes announcing false locations.

4. PROPOSED SYSTEM

In this paper, we focus on the latter aspect, here in after referred to as neighbor position verification (NPV for short).Specifically, we deal with a mobile ad hoc network, where pervasive infrastructure is not present, and the location data must be obtained through node-to-node communication. Such a scenario is of particular interest since it leaves the door open for adversarial nodes to misuse or disrupt the location-based services. For example, by advertising forged positions, adversaries could bias geographic routing or data gathering processes, attracting network traffic and then eavesdropping or discarding it. Similarly, counterfeit positions could grant adversaries unauthorized access to location-dependent services, let vehicles forfeit road tolls, disrupt vehicular traffic or endanger passengers and drivers. We propose a fully distributed cooperative scheme for NPV, which enables a node, hereinafter called the verifier, to discover and verify the position of its communication neighbors. For clarity, here we summarize the principles of the protocol as well as the gist of its resilience analysis. Detailed discussions of message format, verification tests, and protocol resilience are provided in

Sections 5 and 6. A verifier, S, can initiate the protocol at any time instant, by triggering the 4-step message exchange depicted in Fig. 1, within its 1-hop neighborhood. The aim of the message exchange is to let S collect information it can use to compute distances between any pair of its communication neighbors. To that end, POLL and REPLY messages are first broadcasted by S and its neighbors, respectively. These messages are anonymous and take advantage of the broadcast nature of the wireless medium, allowing nodes to record reciprocal timing information without disclosing their identities.

Then, after a REVEAL broadcast by the verifier, nodes disclose to S, through secure and authenticated REPORT messages, their identities as well as the anonymous timing information they collected. The verifier S uses such data to match timings and identities; then, it uses the timings to perform ToF-based ranging and compute distances between all pairs of communicating nodes in its neighborhood. Once S has derived such distances, it runs several position verification tests in order to classify each candidate neighbor as either: 1. Verified, i.e., a node the verifier deems to be at the claimed position; 2. Faulty, i.e., a node the verifier deems to have announced an incorrect position; 3. Unverifiable, i.e., a node the verifier cannot prove to be either correct or faulty, due to insufficient information.

Advantages of Proposed System

Our NPV scheme is compatible with state-of the-art security architectures, including the ones that have been proposed for vehicular networks. It is lightweight, as it generates low overhead traffic. It is robust against independent and colluding adversaries

5. IMPLEMENTATION

NPV Protocol

TABLE 1
Summary of Notations

Notation	*Description*
k_X (K_X)	private (public) key of X
k'_X (K'_X)	private (public) one-time key of X
t_X (t'_X)	actual (fake) transmission time of a message by X
t_{XY} (t'_{XY})	actual (fake) reception time at Y of a message by X
p_X (p'_X)	actual (fake) position of X
d_{XY}	distance between X and Y
ϵ_p (ϵ_r)	position (ranging) error
ϵ_m	tolerance to node movements during protocol execution
R	node proximity range
$\mathbb{N}_X$	current set of X's comm. neighbors
T_X	random wait interval after POLL reception at X
ρ_X	nonce sent by X
Sig_X	digital signature of X
C_X	certificate of X
$\mathfrak{c}_X$	*commitment* of X
i_X	temporary identifier assigned by S to X
$\mathbb{V}_X$	set of *verified* comm. neighbors of X
$\mathbb{U}_X$	set of *unverifiable* comm. neighbors of X
$\mathbb{F}_X$	set of *faulty* comm. neighbors of X
$\mathbb{W}_X$	set of *conditionally verified* comm. neighbors of X

We detail the message exchange between the verifier and its communication neighbors, followed by a description of the tests run by the verifier. Table 1 summarizes the notations used throughout the protocol description.

Protocol Message Exchange

The value pX is the current position of X, and INX is the current set of its communication neighbors. We denote by tX the time at which a node X starts a broadcast transmission and by tXY the time at which a node Y starts receiving it. Note that these time values refer to the actual instant at which the node starts transmitting/receiving the first bit of the message at the physical layer.

Algorithm 1. Message exchange protocol: verifier.

1 **node** S **do**
2 $S \to * : \langle \text{POLL}, K'_S \rangle$
3 S : store t_S
4 **when** *receive* REPLY *from* $X \in \mathbb{N}_S$ **do**
5 S : store $t_{XS}, \mathbb{c}_X$
6 **after** $T_{max} + \Delta + T_{jitter}$ **do**
7 $S : \mathbb{m}_S = \{(\mathbb{c}_X, i_X) \mid \exists\, t_{XS}\}$
8 $S \to * : \langle \text{REVEAL}, \mathbb{m}_S, E_{k'_S}\{h_{K'_S}\}, Sig_S, C_S \rangle$

Algorithm 2. Message exchange protocol: any neighbor.

1 **forall** $X \in \mathbb{N}_S$ **do**
2 **when** *receive* POLL *by* S **do**
3 X : store t_{SX}
4 X : extract T_X uniform r.v. $\in [0, T_{max}]$
5 **after** T_X **do**
6 X : extract nonce ρ_X
7 $X : \mathbb{c}_X = E_{K'_S}\{t_{SX}, \rho_X\}$
8 $X \to * : \langle \text{REPLY}, \mathbb{c}_X, h_{K'_S} \rangle$
9 X : store t_X
10 **when** *receive* REPLY *from* $Y \in \mathbb{N}_S \cap \mathbb{N}_X$ **do**
11 X : store $t_{YX}, \mathbb{c}_Y$
12 **when** *receive* REVEAL *from* S **do**
13 $X : \mathbb{t}_X = \{(t_{YX}, i_Y) \mid \exists\, t_{YX}\}$
14 $X \to S$:
$\langle \text{REPORT}, E_{K_S}\{p_X, t_X, \mathbb{t}_X, \rho_X, Sig_X, C_X\} \rangle$

To retrieve the exact transmission and reception time instants, avoiding the unpredictable latencies introduced by interrupts triggered at the drivers level, a solution such as that implemented in this required.3 Furthermore, the GPS receiver should be integrated in the 802.11 card; software defined radio solutions combining GPS and 802.11 capabilities are proposed, among others, Now, consider a verifier S that initiates the NPV protocol. The message exchange procedure is outlined in Algorithm 1 for S, and in Algorithm 2 for any of S communication neighbors.

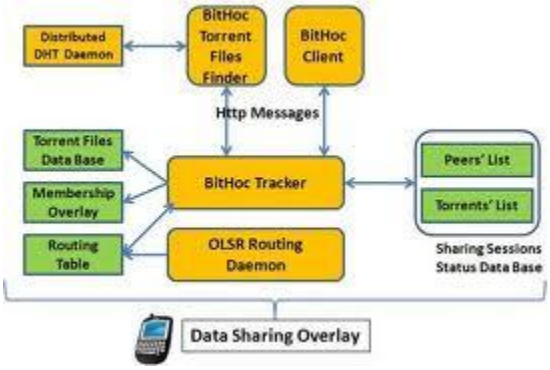
The Direct Symmetry Test (DST)

DST is the first verification performed by S and is detailed in Algorithm 3. There, j _ j denotes the absolute value operator and kpX _ pY k the euclidean distance between locations pX and pY. In the DST, S verifies the direct links with its communication neighbors. To this end, it checks v whether reciprocal ToF-derived distances are consistent 1) with each other, 2) with the position advertised by the neighbor, and 3) with a proximity range R. The latter corresponds to the maximum nominal transmission range, and upper bounds the distance at which two nodes can communicate. More specifically, the first check verifies that the distances dSX and dXS, obtained from ranging, do not differ by more than twice the ranging error plus a tolerance value _m (Algorithm 3, line 4), accounting for node spatial movements during the protocol execution. The second check verifies that the position advertised by the neighbor is consistent with such distances, within an error margin of 2_p þ _r (Algorithm 3, line 5). Although trivial, this check is fundamental since it correlates positions to computed distances: without it, an attacker could fool the verifier by simply advertising an arbitrary position along with correct broadcast transmission and reception timings. Finally, as a sanity check, S verifies that dSX is not larger than R (Algorithm 3, line 6). The verifier tags a neighbor as faulty if a

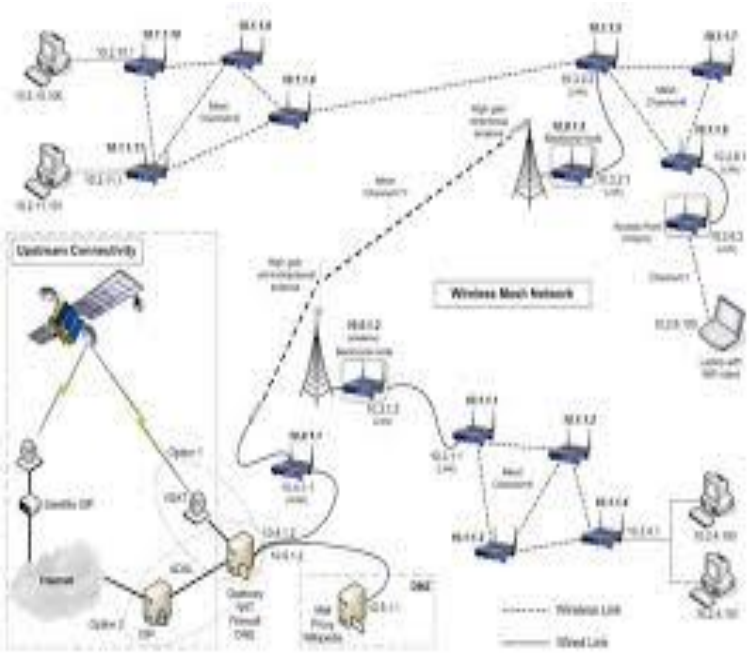
mismatch is found in any of these checks,4 since this implies an inconsistency between the position pX and the timings announced by the neighbor (tSX, tX) or recorded by the verifier (tXS, tS).

6. SYSTEM ARCHITECTURE

When a mobile node moves and starts receiving information messages from multiple GWs, it must select one of them, using criteria that minimize the distance in hops to the GW, and maximize the network stability (or other criteria). Changing GW may imply handover.



The Daidalos nodes directly connected to the infrastructure (1 hop distance) use the Fast Handover mechanism, which provides mobility and very low packet loss probability. The ad-hoc handover process proposed differs from the Fast Handover process: it minimizes the handover related signalling messages (since it traverses multiple nodes) at the expense of some packet loss.



7. CONCLUSION

We presented a distributed solution for NPV, which allows any node in a mobile ad hoc network to verify the position of its communication neighbors without relying on a priori trustworthy nodes. Our analysis showed that our protocol is very robust to attacks by independent as well as colluding adversaries, even when they have perfect knowledge of the neighborhood of the verifier. Simulation results confirm that our solution is effective in identifying nodes advertising false positions, while keeping the probability of false positives low. Only an overwhelming presence of colluding adversaries in the neighborhood of the verifier, or the unlikely presence of fully collinear network topologies, can degrade the effectiveness of our NPV. Future work will aim at integrating the NPV protocol in higher layer protocols, as well as at extending it to a proactive paradigm, useful in presence of applications that need each node to constantly verify the position of its neighbors.

REFERENCES

[1] 1609.2-2006: IEEE Trial-Use Standard for Wireless Access in Vehicular VEnvironments - Security Services for Applications and Management Messages, IEEE, 2006.

[2] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure Vehicular Communications: Design and Architecture," IEEE Comm. Magazine, vol. 46, no. 11, pp. 100-109, Nov. 2008.

[3] P. Papadimitratos and A. Jovanovic, "GNSS-Based Positioning: Attacks and Countermeasures," Proc. IEEE Military Comm. Conf. (MILCOM), Nov. 2008.

[4] L. Lazos and R. Poovendran, "HiRLoc: High-Resolution Robust Localization for Wireless Sensor Networks," IEEE J. Selected Areas in Comm., vol. 24, no. 2, pp. 233-246, Feb. 2006.

[5] R. Poovendran and L. Lazos, "A Graph Theoretic Framework for Preventing the Wormhole Attack," Wireless Networks, vol. 13, pp. 27-59, 2007.

[6] S. Zhong, M. Jadliwala, S. Upadhyaya, and C. Qiao, "Towards a Theory of Robust Localization against Malicious Beacon Nodes," Proc. IEEE INFOCOM, Apr. 2008.

[7] P. Papadimitratos, M. Poturalski, P. Schaller, P. Lafourcade, D. Basin, S. _Capkun, and J.-P. Hubaux, "Secure Neighborhood Discovery:AFundamental Element for Mobile Ad Hoc Networks," IEEE Comm. Magazine, vol. 46, no. 2, pp. 132-139, Feb. 2008.

[8] Y.-C. Hu, A. Perrig, and D.B. Johnson, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks," Proc. IEEE INFOCOM, Apr. 2003.

[9] J. Eriksson, S. Krishnamurthy, and M. Faloutsos, "TrueLink: A Practical Countermeasure to the Wormhole Attack in Wireless Networks," Proc. IEEE 14th Int'l Conf. Network Protocols (ICNP), Nov. 2006.

[10] R. Maheshwari, J. Gao, and S. Das, "Detecting Wormhole Attacks in Wireless Networks Using Connectivity Information," Proc. IEEE INFOCOM, Apr. 2007.

[11] R. Shokri, M. Poturalski, G. Ravot, P. Papadimitratos, and J.P. Hubaux, "A Practical Secure Neighbor Verification Protocol for Wireless Sensor Networks," Proc. Second ACM Conf. Wireless Network Security (WiSec), Mar. 2009.

[12] M. Poturalski, P. Papadimitratos, and J.-P. Hubaux, "Secure Neighbor Discovery in Wireless Networks: Formal Investigation of Possibility," Proc. ACM Symp. Information, Computer and Comm. Security (ASIACCS), Mar. 2008.

[13] M. Poturalksi, P. Papadimitratos, and J.-P. Hubaux, "Towards Provable Secure Neighbor Discovery in Wireless Networks," Proc. Workshop Formal Methods in Security Eng., Oct. 2008.

[14] E. Ekici, S. Vural, J. McNair, and D. Al-Abri, "Secure ProbabilisticLocation Verification in Randomly Deployed Wireless Sensor Networks," Elsevier Ad Hoc Networks, vol. 6, no. 2, pp. 195-209, 2008.

[15] J. Chiang, J. Haas, and Y. Hu, "Secure and Precise Location Verification Using Distance Bounding and Simultaneous Multilateration," Proc. Second ACM Conf. Wireless Network Security (WiSec), Mar. 2009.

[16] S. _Capkun, K. Rasmussen, M. Cagalj, and M. Srivastava, "Secure Location Verification with Hidden and Mobile Base Stations," IEEE Trans. Mobile Computing, vol. 7, no. 4, pp. 470-483, Apr. 2008.

[17] S. _Capkun and J.-P. Hubaux, "Secure Positioning in Wireless Networks," IEEE J. Selected Areas in Comm., vol. 24, no. 2, pp. 221- 232, Feb. 2006.

[18] A. Vora and M. Nesterenko, "Secure Location Verification Using Radio Broadcast," IEEE Trans. Dependable and Secure Computing, vol. 3, no. 4, pp. 377-385, Oct.-Dec. 2006.

[19] J. Hwang, T. He, and Y. Kim, "Detecting Phantom Nodes in Wireless Sensor Networks," Proc. IEEE INFOCOM, May 2007.

AUTHORS' BIOGRAPHY

Anil Kumar Gona is a student pursuing M.Tech(CSE) in Eswar College of Engineering, Narasaraopet, Guntur, India. He also Completed MCA.

Ratna Raju Mukiri M.Tech(CSE), S.E.T.,(P.hD)., is having 10+ years of experience in the field of teaching in various Engineering Colleges and PG colleges. At present he is working as Asst. Prof. in Eswar College of Engineering, Narasaraopet, Guntur, India. He published 4 international journals and attend 1 national conference and 1 international conference and qualified state eligibility test twice in 2012 & 2013. He has given many guest lecturers to M.C.A. students in the subject areas of Micro processors, artificial intelligence, data structures etc., He also guided many **B.Tech**, **MCA** and **M.Sc(CS)** projects. He attended two weeks **ISTE workshop** on "**Data Base Management Systems**" conducted by **IIT Bombay**. His interested areas are data mining, mobile computing software engineering, Computer Networks, etc.