IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 1, Ver. VIII (Feb. 2014), PP 24-31
www.iosrjournals.org

Dissemination of Link State Information for Enhancing Security in Mobile Ad Hoc Networks

Chandrasekar P[1], BeaulahDavid[2] ,Shaheen H[3]
PG Scholar Nehru Institute of Technology Coimbatore-641105 Tamilnadu
Assistant Professor Nehru Institute of Technology Coimbatore-641105 Tamilnadu
Assistant Professor Nehru Institute of Engineering and Technology Coimbatore-641105 Tamilnadu

Abstract: *A mobile adhoc network is a Self-configuring network of mobile routers connected by wireless links. In the mobile adhoc network, each and every device moves independently in any direction so that there are frequent changes in the links. It is essential to learn the position of the neighbors because there is increase in location-aware services. So, there is a chance that the malicious nodes are easily abused the process. The significant problem in mobile networks is correctness of node locations and also it is primarily challenging in the presence of adversaries. So, the neighbor position verification protocol is used to a fully distributed, a lightweight NPV procedure which allows each node to obtain the locations advertised by its neighbors and asses their truthfulness. Further to extend neighbor position verification protocols in the proactive model that need to each node constantly verify the position of its neighbors. So, we introduce a technique called secure link state updating which provides secure proactive topology discovery that is multiply useful for the network operation. This technique is vigorous against individual attackers, it is capable to adjust its capacity between local and network-wide topology discovery, and also operating in networks of frequently changing topology and membership nodes. Experimental results show that the proposed system is high efficiency in terms of security when compared to the existing system.*

Key Terms:*Mobile adhoc networks, Neighbor position verification, location-aware service, link state updation*

I. Introduction

A mobile adhoc network is a type of adhoc network which can change locations and configure itself on the fly. The mobile adhoc network is an autonomous collection of mobile users that communicate over relatively bandwidth constrained wireless links. The nodes in the mobile adhoc network are mobile so that the network topology changes rapidly and unpredictably over time. Location awareness has become a positive feature in mobile systems in which a wide range of applications need the position of the participating nodes. Some applications require the neighbor position information such as geographic routing in spontaneous networks, data collection in sensor networks, location-specific services for handheld devices and traffic monitoring in vehicular networks.

In the mobile adhoc networks, due to the dynamic changes in the network topology, there is a presence numerous security attacks which can actively disrupt the routing protocol and disable communication. Privacy, misbehavior of the nodes, insecure neighbor discovery is the significant aspects of the mobile adhoc networks. In these situations there is necessitate to establish the correct location in spite of attacks feeding false location information and to verify the positions of the neighbors so that to detect the adversarial nodes announcing false locations. In the mobile adhoc network, when a node wants to forward the data it chooses the neighbor node which is nearest to the destination. Malicious nodes can deliberately lie about their positions for several reasons. A malicious node can be selected as an intermediate relay node by manipulating its own position information. It can then either drop the data packets or change the content of the packets.

In the wireless networks, neighbors are typically defined as nodes that are in the within radio range. Neighbor discovery is the process in which a node in a network decides the total number and identity of other nodes in its vicinity. A neighbor position verification (NPV) is used to discover and verify the position of the neighbors. Particularly, in a mobile adhoc network, where a pervasive infrastructure is not present, and the location data must be attained through node-to-node communication. This situation is a chance for the adversarial nodes to misuse the location-based services. By advertise the forged positions, adversaries could bias data gathering processes, attracting network traffic and then discard the data. Further to integrate the NPV protocol in higher layer protocols as well as extend it to a proactive paradigm and it is useful in the presence of applications in which every node constantly verifies the position of its neighbors. So, in the proposed system, a secure link state updation is utilized. This method is used to provide secure proactive routing that secures the discovery and the distribution of link state information across mobile adhoc domains.

II. Related Work

PanagiotisPapadimitratos et.al suggested analyzing the vulnerability and effectiveness of the Global Navigation Satellite Systems [1]. Mostly, mobile devices attain their own location with the help of Global Navigation Satellite systems for example a Global Positioning System (GPS) receiver. By attacking the GNSS-based positioning, it can counterfeit navigation messages and misinform the receiver into manipulating a fake location. For analyzing the vulnerability, firstly to consider replay attacks that can be effective in the presence of future cryptographic GNSS protection mechanisms. After that, propose the methods that permit GNSS receivers to detect the reception of signals generated by an adversary and then reject fake locations. To consider three diverse mechanisms based on own location time, and Doppler shift, receivers can attain prior to the onset of an attack. But this method only partially limits the impact of the attack.

LoukasLazos et.al proposed High resolution Range-independent Localization scheme in the wireless sensor networks [2]. This method permits sensors to passively decide the location with high accuracy. This method cannot enlarge the complexity of the hardware of each and every reference point. In the High resolution Range-independent Localization scheme sensors decide their location based on the intersection of the areas covered by the beacons transmitted by multiple reference points. The increased localization accuracy is the result of combination of multiple localization information over a short time period, and does not come at the cost ofincreased hardware complexity. This method does not perform any range measurements to compute the sensors location; it is not susceptible to any range measurement alteration attacks. The High resolution Range-independent Localization leads to significant improvement in localization accuracy with fewer hardware resources.

RadhaPoovendran et.al suggested a graph theoretic framework for preventing the Wormhole attack in the wireless adhoc network [3]. To investigate the wormhole attack in wireless adhoc network, an attack that can interrupt vital network functions such as routing. In wormhole attack, an attacker creates a low-latency unidirectional or bi-directional wired or wireless link between the two points in the network. A graph theoretic framework is used for modeling wormhole links and derives the essential and sufficient conditions for detecting and defending against wormhole attacks. Based on this framework, the solution should prevent wormholes should construct a communication graph that is a sub graph of the geometric graph defined by the radio range of the network nodes. By using the framework, a cryptographic mechanism is proposed based on local broadcast keysin order to prevent wormholes. This method does not need time synchronization, requires only a small fraction of the nodes to know their location, and is decentralized. But this method does not provide efficient security for detecting the wormhole attacks.

Yih-Chun Hu et.al proposed a mechanism called packet leashes for detecting and defending against wormhole attacks [4]. Security is an essential requirement in the mobile adhoc networks. The wormhole attack is a severe attack in adhoc networks which is particularly challenging to defend against. In a wormhole attack, attacker records packets at one location, tunnel them to another location and retransmits them there into the network. The wormhole attack can form a severe threat in wireless networks, particularly against many ad hoc network routing protocols and location-based wireless security systems. To introduce the general mechanism of packet leashes to detect wormhole attacks and to present two types of leashes: geographic leashes and temporal leashes.A geographical leash makes sure that the recipient of the packet is within a certain distance from the sender. A temporal leash makes sure that the packet has an upper bound on its lifetime that limits the maximum travel distance, because the packet can travel at most at the speed of light. Either type of leash can avoid the wormhole attack, because it permits the receiver of a packet to sense if the packet traveled further than the leash allows. But this makes more complexity and computation time.

Jakob Eriksson et.al suggested True link timing based countermeasure to the wormhole attach in wireless networks [5]. In a wormhole attack, wireless transmissions are recorded at one location and replayed at another creating a virtual link under attacker control. True link performs link verification between two nodes I and j in two phases. One is the rendezvous phase, and the one is authentication phase. In a rendezvous phase, i and j exchange noncesα_j and β_i in which the subscript denotes that the node that generated the nonce. This exchange establishes the adjacency of the responding node through the use of strict timing constraints; only a direct neighbor is capable to respond in time. In an authentication phase, i and j every node sign and transmit the message mutually authenticating themselves as the originator of their respective nonce.

RiteshMaheshwari et.al proposed a novel algorithm for identifying the wormhole attacks in wireless networks [6]. This novel algorithm utilizes only connectivity information to look forbidden substructures in the connectivity graph. This approach is totally localized and does not use special hardware, making the technique generally applicable. The wireless communication model between the nodes is used in the detection algorithm. Because a communication model can help define what substructures observed in the connectivity graph could be forbidden. On the other hand, this approach is applicable when the communication model is unknown. In a wormhole detection algorithm, starting from the unit disk graph model and then general communication models,

and to analyze and automatically remove links created by wormhole once a wormhole is detected. But this method does not provide efficient security.

MarcinPoturalski et.al suggested secure neighbor discovery in wireless networks. Wireless communications are used in a wide spectrum of applications that ranging from product to strategic systems [7]. Neighbor discovery is the method which decides which devices are within direct a radio communication, that is a building block of network protocols and applications, and its susceptibility can rigorously compromise their functionalities. In this method a formal model is build to capture the most important features of wireless systems, most remarkably obstacles and interference and to provide a specification of a basic variant of the neighbor discovery problem. Then derive an unfeasibility resultfor a common class of protocols we term "time-based protocols," to which many of the schemes in the literature belong. Also recognize the conditions under which the unfeasibility result is lifted. Furthermore, to discover a second class of protocols we term "time- and location-based protocols," and prove they can secure ND. But in this method does not give importance to the wide spectrum of protocols and also does not control the transmission power.

J. McNair et.al proposed position information confirmation system for wireless sensor networks. In the wireless sensor networks security plays an important function to organize and retrieve trustworthy data [8]. Position verification is an effectual technique that resistance against attacks that take benefit of a lack, or compromise, of location information. A probabilistic position confirmation method is used for arbitrarily deployed dense sensor networks. The proposed Probabilistic position confirmation (PLV) algorithm leverages the probabilistic dependence of the number of hops a broadcast packet which traverses to attain a destination and the Euclidean distance between the source and the destination. A small number of verifier nodes are utilized to decide the plausibility of the claimed location, which is signified by a real number between zero and one. By utilizing the calculated plausibility metric, it is probable to generate arbitrary number of trust levels in the location claimed. But this method does not sustain for the position of one end of the tunnel can be identified by tracing the hop count gradient in reverse.

III. Neighbor Position Verification

Mobile adhoc networks are vulnerable to attacks so as to detect the adversaries Neighbor position verification method is used. In the occurrence of attackers to require solutions that let nodes to properly launch their location in spite of attacks feeding false position information and validate the locations of their neighbors, so as to distinguish adversarial nodes announcing false locations. In particular, neighbor position verification is deal with a mobile adhoc network, in which a pervasive infrastructure is not present, and the location data must be attained via node-to-node communication. Some of the characteristics of neighbor position verification. 1) It is considered for impulsive ad hoc environments, and, also it does not rely on the occurrence of a trusted infrastructure. 2) This protocol is reactive i.e., that can be executed by any node, at any point of time without any knowledge of the neighborhood. 3) It is vigorous against independent and colluding attackers. 4) It is lightweight, as it creates low overhead traffic.

3.1 Cooperative NPV

In the neighbor Position verification protocol a fully distributed cooperative scheme is utilized which ensures a node i.e., verifier. The communications neighbors are discovered and validate it. A verifier instigates the protocol by generating the 4-step message exchange within its 1-hop neighborhood. The objective of the message exchange is that the verifier collects information it can use evaluate distances between any pair of its communication neighbors. The POLL and REPLY messages are broadcasted by the verifier and its neighbors, permitting nodes to record mutual timing information without disclosing their identities. After that a REVEAL broadcast by the verifier, nodes reveal to verifier through secure and authenticated REPORT messages, their identities as well as the unidentified timing information they collected. The verifier utilizes the data to match timings and identities and by using the timings to perform Time-of-flight based ranging and compute distances between all pairs of communicating nodes in its neighborhood. Once the verifier computes the distances it runs several verification tests in order to categorize the candidate.1) Verified: the verifier estimates that the node is at the claimed position. 2) Faulty: the verifier estimates that the node is in an incorrect position. 3) Unverifiable: the verifier estimates that the node is either correct or faulty.

3.1.1 Message Exchange process

The $P_X$is the current position of X , and the current set of its communication neighbors is denoted asIN_X. t_X denotes the time at which a node X starts a broadcast transmission and by t_{XY} the time at which a node Y starts receiving it. The message exchange process is shown in algorithm 1 and 2.

1. node S
2. S $\rightarrow *: \prec POLL, K_S^{'} \succ$
3. S : store t_S

4. When receive REPLY from X $\ll \epsilon \mathbb{N}_S do$
5. S : store $t_{XS}, \mathbb{C}_X$
6. After $T_{max} + \Delta + T_{jitter} do$
7. S : $m_S = \{(\mathbb{C}_X, i_X) | \exists t_{XS}\}$
8. S $\rightarrow * : \prec REVEAL, m_S, E_{K_S'}\left\{h_{K_{SS}'}\right\}, Sig_S, C_S \succ$

Algorithm1. Message exchange protocol: verifier

1. For all X $\epsilon \mathbb{N}_S$ do
2. When receive POLL by S do
3. X: store t_{SX}
4. X : extract T_X uniform r.v ϵ $[0, T_{max}]$
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 $\rightarrow *: \prec REPLY, \mathbb{C}_X, h_{K_S'} \succ$
9. X : store t_X
10. When receive REPLY from Y $\epsilon \mathbb{N}_S \cap N_X$ do
11. X : store $t_{YX}, \mathbb{C}_Y$
12. When receive REVEAL from S do
13. X: $\mathbb{t}_X = \{t_{YX}, i_Y) | \exists t_{YX}$
14. X $\rightarrow S: \prec REPORT, E_{KS}\{p_X, t_X, \mathbb{t}_X, \rho_X, Sig_X, C_X\} \succ$

Algorithm2. Message exchange protocol: any neighbor

3.1.1.1 ***POOL Message:***

At thetransmission timet_S , Verifier Stransmit a POOL message and it saves locally. The POLL is anonymous, it does not take the uniqueness of the verifier, it is transmitted employing a fresh, software-generated MAC address, and it includes a public key K_S' taken from S's pool of anonymous one-time use keys that do not permit neighbors to map the key onto a specific node.

3.1.1.2 ***REPLY message*****:**

The POOL message and its reception time is received by the communication neighbor if XϵIN_S and extort a random wait interval $T_X \epsilon$ [0,T_{max}]. After the random time interval has elapsed X transmits an anonymous REPLY message by using a fresh MAC address, and locally records its transmission timet_X. There are some information including REPLY message like encrypted with S public key, the reception time of POOL message and a nonce ρ_X used to tie the REPLY to the next message sent by X: we refer to these data as X's commitment, C_X. If a neighbor X sends a REPLY message, the verifier S stores the reception time t_{XS} and the commitmentC_X.

3.1.1.3 ***REVEAL message:***

The verifier broadcasts a REVEAL message by using its real MAC address after a time T_{max} + Δ +T_{jitter}. Δ denotes the propagation and contention lag of REPLY messages scheduled at time T_{max} and T_{jitter} is a random time added to thwart jamming efforts on this message. The REVEAL message includes: 1) a mapIm_S, that correlates each commitment C_X received by the verifier to a temporary identifieri_X; 2) S is the author of the original POLL through the encrypted hash $E_{k_s'}\{h_{k_s'}\}$. 3) The verifier identity which denotes that its certified public key and signature.

3.1.1.4 ***REPORT message*****:**

In the REPORT message, it takes information like position of X, communication time of X's REPLY, and the list of pairs of reception times and short-term identifiers referring to the REPLY broadcasts X received. In the REVEAL message, the identifiers are attained from the map Im_S are also included. X reveals its own identity by containing its digital signature and certified public key: through the nonceρ_X, it associates the REPORT to its previously issued REPLY. By using the S's public key,K_S, all the sensitive data are encrypted, so that eavesdropping on the wireless channel is not possible. Finally, only the verifier recognizes the locations and information timings.

3.2 Verification tests

3.2.1 Direct Symmetry Test

In this verification test, the Euclidean distance between the p_X and $p_Y$is denoted by $||p_{X-}p_Y||$and |.| indicates that the absolute value operator. The direct links with its communiqué neighbors is verified by the verifier S. After that, it checks whether reciprocal ToF-derived distances are reliable 1) with each other, 2) with the position presented by the neighbor, and 3) with a closeness range R. Particularly, the verifier validates that the distances d_{SX} andd_{XS}, attained from ranging do not differ by more than twice the ranging error plus a tolerance valueϵ_m, accounting for node spatial movements during the protocol execution. In the second check, validates that the position advertised by the neighbor is consistent with such distances, within an error margin of $2\epsilon_p$ +ϵ_r. At the end, as a sanity check, Verifier verifies that d_{SX} is not superior than R. The verifier tags a neighbor as defective if a difference is found in any of these checks, because this implies an irregularity between the position p_X and the timings announced by the neighbor $(t_{SX,}t_X)$ or recorded by the verifier(t_{XS},t_X).

1. Node S do
2. S : $\mathbb{F}_S \leftarrow \emptyset$
3. For all X $\epsilon \mathbb{N}_S do$
4. If $|d_{SX} - d_{XS}| > 2\epsilon_r + \epsilon_m or$
5. $\big||p_S - p_X|\big| - d_{SX}| > 2\epsilon_p + \epsilon_r or$
6. $d_{SX} > R then$
7. S : $\mathbb{F}_S \leftarrow X$

Algorithm 3. Direct Symmetry Test

3.2.2 Cross-Symmetry Test

In the Cross-Symmetry Test, the information collected by the communiqué neighbors is validated. This test believes the nodes that verified to be communication neighbors between each other, i.e., for which Time-of-Flight -derived mutual distances are obtainable. On the other hand, pairs of neighbors declaring collinear positions with respect to S are not taken into consideration. This test validates the regularity of the reciprocal distances for all other pairs of neighbors. For each and every neighbor X, S maintains a link counter l_X and a mismatch counterm_X. The former is incremented at every new crosscheck on X, and records the number of links between X and other neighbors of S. At every time, it is incremented at least one of the cross-checks on distance and position fails, and to discover the potential for X being defective.

1. Node S do
2. S : $\mathbb{U}_S \leftarrow \emptyset, \mathbb{w}_S \leftarrow \emptyset$
3. For all X $\epsilon \mathbb{N}_S, X \notin \mathbb{F}_S do$
4. S : $l_X = 0, m_X = 0$
5. For all (X,Y) | X,Y $\epsilon \mathbb{N}_S, X, Y \notin \mathbb{F}_S, X \neq Y do$
6. If $\exists d_{XY}, d_{YX} and$
7. $\rho_S \notin line(p_X, p_Y) then$
8. S : $l_X = l_X + 1, l_Y = l_Y + 1$
9. If $|d_{XY} - d_{YX}| > 2\epsilon_r + \epsilon_m or$
10. $|||p_X - p_Y||-d_{XY}| > 2\epsilon_r + \varepsilon_m or$
11. $d_{XY} > R then$
12. S: $m_X = m_X + 1\,, m_Y = m_Y + 1$
13. For all X $\epsilon \mathbb{N}_S, X \notin \mathbb{F}_S do$
14. If $l_X < 2\ then S$: $\mathbb{U}_S \leftarrow X$
15. Else switch $\frac{m_X}{l_X} do$
16. Case $\frac{m_X}{l_X} > \delta S$: $\mathbb{F}_S \leftarrow X$
17. Case $\frac{m_X}{l_X} = \delta S$: $\mathbb{U}_S \leftarrow X$
18. Case $\frac{m_X}{l_X} < \delta S$: $\mathbb{W}_S \leftarrow X$

Algorithm 4. Cross-Symmetry Test (CST)

3.2.3 Multi-lateration Test

In this type of test, the defective and the unverifiable nodes are ignored. For each and every neighbor X that did not inform about a link accounted by another node Y, with X, Y ϵW_s a curve L_X(S, Y) is evaluated and added to the set IL_X. The generated curve is the locus of points which creates a transmission, whose time Difference of Arrival (TDoA) at S and Y matches which computed by the two nodes. It is simple to validate that such a curve is a hyperbola, with foci in p_S and p_Y , and passing through the actual position of X.

1. Node S do
2. S : $\mathbb{v}_S \leftarrow \emptyset$

3. For all X $\epsilon \mathbb{W}_S do$
4. S : $\mathbb{L}_X \leftarrow \emptyset$
5. For all (X,Y)| X,Y $\epsilon \mathbb{W}_S, X \neq Y do$
6. If $\exists t_{XY} and \nexists t_{YX} then$
7. S : $\mathbb{L}_X \leftarrow L_X(S,Y)$
8. For all X $\epsilon \mathbb{W}_S do$
9. If $|\mathbb{L}_X| \geq 2\ then$
10. S : $p_X^{ML} = argmin_p \sum_{L_i,L_j \epsilon \mathbb{L}_X} ||p - L_i \cap L_j||^2$
11. If $\|p_X - p_X^{ML}\| > 2\epsilon_p then$
12. S : $\mathbb{F}_S \leftarrow X, \mathbb{W}_S = \mathbb{W}_S \backslash X$
13. S: $\mathbb{V}_S = \mathbb{W}_S$

Algorithm 5. Multi-lateration Test (MLT)

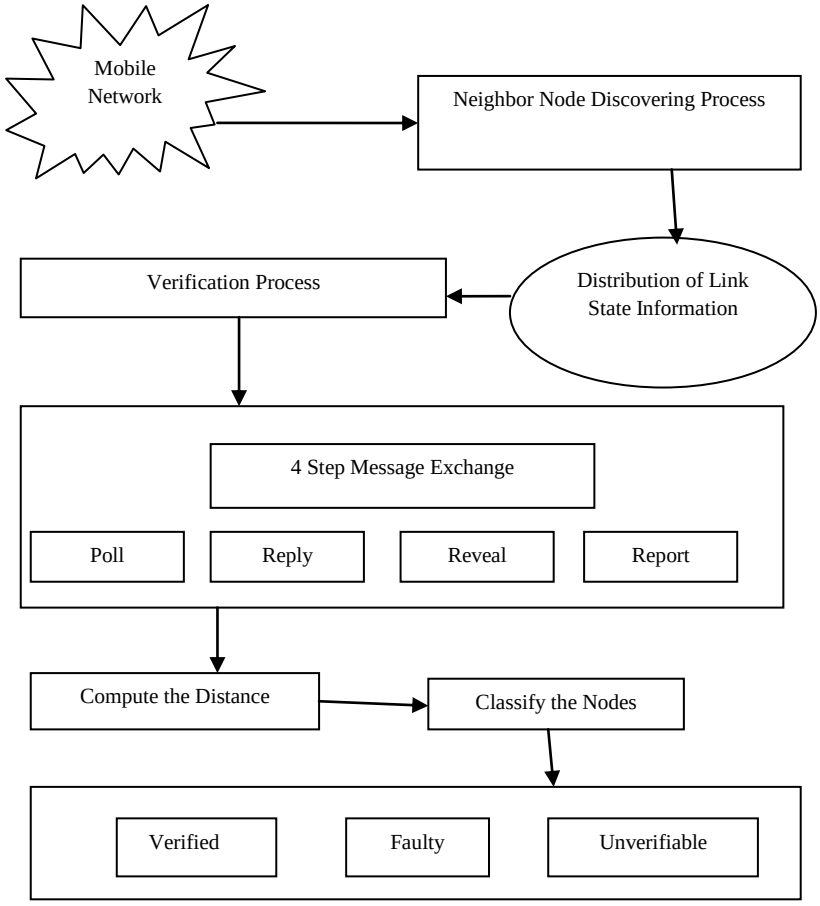

Figure 1. Architecture Diagram

IV. Distribution Of Link State Information

To enhance the security the dissemination of link state information method is used. In this method, to propose a proactive MANET protocol which secures the discovery and the allotment of link state information across mobile ad hoc domains. The main objective is to provide correct, up-to-date, and authentic link state information, robust against Byzantine behavior and failures of individual nodes. The choice of a link state protocol provides such robustness, unlike distance vector protocols , which can be considerably more affected by a single misbehaving node. In addition to that, the accessibility of explicit connectivity information, present in link state protocols, has extra benefits: for example it contains the capability of the source to decide and route concurrently across multiple routes, the operation of the local topology for proficient distribution of data or resourceful propagation of control traffic.

Link state updating method provides each node to disseminate its public key to nodes within its zone. Nodes occasionally transmit their certified key, so that the receiving nodes authenticate their subsequent link state updates. If there is changes in the network topology, nodes find out the keys of nodes that move into their zone, thus keeping track of a comparatively limited number of keys at every instance. Nodes

occasionally broadcast their certified key, so that the receiving nodes validate their subsequent link state updates. By broadcasting occasionally signed link state updates, nodes advertise the state of their incident links. This method restricts the propagation of the link state updates packets within the zone of their origin node. Receiving nodes authenticate the updates, repress duplicates, and relay formerly unseen updates that have not already propagated *R* hops. Link state information obtained from validated link state updates packets is established only if both nodes incident on each link advertise the same state of the link.

V. Experimental Results

In this section the experimental results is shown for the existing and the proposed system. In the existing method, neighbor position verification protocol is used to for the neighbor discovering and verification in the mobile adhoc networks. In the proposed system, the distribution of link state information method is used. In the experimental analysis, the adversary decision on the kind of attack to launch is driven by the tradeoff between the chances of success and the freedom of choice on its fake position. In the basic attack, the attacker permits selects any false position, but it needs a high percentage of colluders in the neighborhood in order to be successful. The hyperbola-based attack involves that the less freedom of choice but has higher chances of success. The collinear attack joins the attacker into an exact angle with the verifier, and severely bounds its distance from the verifier itself.

Figure 2. Shows the Average displacement allowed in the various types of attacks. This figure compares the displacement values for both the existing and the proposed method. To examine that victorious collinear attacks defer small benefit for attackers, who are required to proclaim locations quite close to their real locations. To terminate those collinear attacks, usually those with the highest chances of success as formerly discussed are also those resulting in the smallest gain for the adversaries. On the other hand, basic attacks allow the largest average for

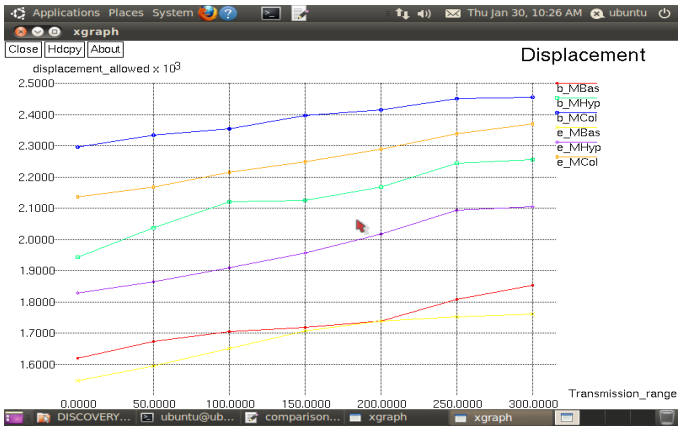

Figure 2. Average Displacement

displacements, but we showed that they have extremely low success probability the hyperbola-based attacks appear then to be the most dangerous ones, if the displacement gain is taken into consideration.

Figure 3. Traffic load per verification

Figure 3 shows that the traffic loads per verification for the existing and the proposed system. The plot only considers for transmission range differences since, once more, the other parameters do not have an force on the overhead. To examine that security comes at a cost, because the traffic load of the neighbor position verification protocol is superior to that of a basic non-secure neighbor position discovery, consisting of only one poll and associated position replies from neighbors. More accurately, the neighbor position verification protocol

overhead is similar to that of the non-secure discovery for lesser transmission ranges, at the same time as the difference tends to increase for larger ranges. Compared to the existing system, in the proposed system, the traffic load per verification is decreases.

VI. Conclusion

A distributed solution for Neighbor Position Verification, that permits any node in a mobile ad hoc network to validate the location of its communication neighbors without relying on a priori trustworthy nodes. This protocol is very vigorous to attacks by independent as well as colluding adversaries, even when they have perfect knowledge of the neighborhood of the verifier. It is effectual in identifying nodes advertising false positions, while keeping the probability of false positives low. In order to improve the security, a secure link state updating method is used for securing the discovery and provides the link state updating information. For future work, in order to find the malicious and selfishly behaving nodes, the trust based security system is developed, This method aims to decrease the number of time slots needed to discover all the neighbors in the network and also it provides security mechanism to improve the cooperation among the neighbor nodes.

References

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

[2] 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.

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

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

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

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

[7] 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.

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