Secure verification of neighborhood membership (SVNM) is a relatively recently stated problem that has an increasing number of practical applications. The problem can be stated as the capability of a wireless network node (verifier) to verify the claim by another node (claimer) that it exists within a certain physical distance from the verifier. Several physical properties of the received signal are used for one-hop location estimation – signal strength, time of flight, and angle of arrival, etc. However, many of these approaches are either not practically applicable or prohibitively expensive in many wireless network scenarios. In this work, we propose algorithms to detect and isolate nodes that lie about their location in the incremental addition phase of static sensor networks. Every node should announce its position and the power level it uses for transmission. Cooperative and base station verifications are used to detect nodes that lie about their locations. Our results show that we can achieve perfect detection (100%) of nodes that falsify their location information using either high power transmission or with the help of other nodes.

Keywords: Multihop wireless networks, security, location verification, location determination.

1 INTRODUCTION

Secure verification of neighborhood membership (SVNM) is a relatively recently stated problem that has an increasing number of practical applications. Secure neighborhood membership verification can be looked upon as a subset of the problem of location determination under the condition that the location of a node can be determined by other nodes. The problem can be stated as the capability of a wireless network node (the verifier) to verify the claim by another node (the claimer) that it exists within a certain physical distance from the verifier. Several physical properties of the received signal are used for one-hop location estimation – signal strength, time of flight, and angle of arrival, etc. However, many of these approaches are either not practically applicable or prohibitively expensive in many wireless network scenarios. Moreover, the location determination protocols typically have an explicit localization phase when beacon messages are exchanged after which each node determines its relative location with respect to its neighbors. This is not secure since a powerful adversary can increase its transmission power for this phase.

In this paper we propose a technique to mitigate the problem of secure location verification using cooperative monitoring and trusted base station. The claimer is supposed to announce the power level he is using and his claimed location. Any node that overhears the announcement of the claimer (called verifier) computes the distance to the claimer using his current location and the location presented in the claimer announcement. Then, the verifier checks whether the power level, within certain thresholds and environmental conditions, supports the computed distance. If the power level is not sufficient, it would be a clue that the current claimer is lying about his location. The first challenging problem that needs to be addressed here is the possibility of colluding among a group of malicious nodes to fake a location of some other legitimate nodes. A second challenging problem arises from the fact that nodes do not trust each other in ad hoc wireless networks scenarios. Therefore, we propose a mechanism to mitigate the trust problem among these nodes.

Recent work [14]-[16], among others, have assumed that the time taken to compromise a sensor node is greater than the time required for neighbor discovery. Since the time taken to compromise a sensor node and the time required for neighbor discovery are both expected to be of the order of seconds, there is a chance that a very small fraction of the nodes are compromised before they perform neighbor discovery. Neighbor discovery, if not correctly performed, can lead to the launch of serious security attacks against the network. For instance, an adversary who wants to unleash a wormhole or a sinkhole attack [17] will want to make his neighbors believe that he lies on the best routing path. Once he succeeds in this operation, he gains control over the routing path and can selectively forward or drop packets, tunnel them to another adversary, etc.

We propose a secure one-hop neighbor verification protocol that enables secure neighbor discovery for incremental node deployments in static multihop wireless sensor networks. Neighbor discovery protocols are vulnerable to a variety of attacks that could either prevent two neighboring nodes from becoming neighbors or could make two non-neighboring nodes to believe that they are neighbors. Our protocol focuses on the latter issue. We show how the adversary can use a special form of the wormhole attack and make two nodes that are not within the communication range of each other to believe that they are
neighbors and show how our protocol effectively counters such an attack.

2 RELATED WORK

The problem of secure neighbor verification is tightly coupled with the problem of secure location verification. Sastry et al. [3] define the problem of secure location verification so that it can be formally treated. A number of researchers commented on the importance of location verification in wireless sensor networks [3]-[4]. Many other researchers (e.g., [7]) have used secure location verification for securing multihop wireless networks. They propose robust mitigation techniques against a wide range of attacks including wormhole, Sybil and rushing attacks.

Several physical properties of the received signal are used for one hop location estimation—signal strength, time of flight, and angle of arrival. Particularly, Hu et al. use temporal packet leashes [5] and Brands et al. use a time-bounded challenge-response protocol [6]. The main limitation of these schemes is the necessity of non-RF ranging hardware which increases the costs of the sensor nodes. Another possible limitation is the need for accurate timing measurements capabilities which may not be available. Other approaches have also been proposed for location verification such as the location-limited channels used by Balfanz et al. [4]. However, the lack of location-limited channels may abridge the suitability of this method. A number of other proposals have also addressed the problem of location and distance estimation for wireless networks (e.g., [4] & [9]). However, they all studied the problem in non-adversarial settings. Distance estimation and positioning techniques are, highly vulnerable to attacks from compromised nodes and external attackers. Compromised nodes can report false position and distance information to cheat on their locations. External attackers can modify the measured positions and distances of wireless nodes.

On the other hand, few proposals for secure distance and location verification have been proposed. Brands et al. [10] propose a distance bounding protocol that can be used to verify the proximity of two devices connected by a wired link. Sastry et al. [3] propose a new distance bounding protocol, based on ultrasound and radio wireless communication. Both proposals focused on the verification of the distance to a device, or on its presence in a region of interest. Lazos et al. [2] proposed a set of techniques for secure positioning in sensor networks based on directional antennas. Kuhn [11] proposed an asymmetric security mechanism for navigation signals. Both proposals address secure location determination by a node, but not secure location verification. Typically the location determination protocols (e.g., [1], [2], and [10]) have an implicit localization phase when beacon messages are exchanged after which each node determines its relative location with respect to its neighbors. However, this is not secure since a powerful adversary can increase its transmission power for just this phase. Moreover, the plethora of existing protocols for a node to determine its own location (e.g. [1]), sometimes in the presence of malicious beacon nodes [2], are asymmetric to our problem where the determination has to be done securely by the neighbors of a node.

3 FOUNDATIONS

3.1 Attack Model and System Assumptions

Attack model: An attacker can control an external node or an internal node, which, since it possesses the keys, can be authenticated by other nodes in the network. An insider node may be created, for example, by compromising a legitimate node. A malicious node can lie about its location or transmission power. An attacker node tries to forge its real location by itself or by colluding with other nodes. However, we do not consider physical-layer jamming [12] or identity spoofing [11]. A malicious node can be more powerful than a legitimate node and can establish out-of-band fast channels (e.g., a wireline link) or have high-powered controllable transmission capability.

System assumptions: We assume that all the legitimate communication links are bi-directional. We assume a key management protocol, e.g., [13], exists such that any two nodes can authenticate their communication.

3.2 Importance of Neighbor Verification

Neighbor verification is essential for almost every routing protocol, MAC protocols, and several other topology-control algorithms such as construction of minimum-energy spanning trees. Neighbor verification is, therefore, a crucial first step in the process of self-organization of wireless sensor networks. Recently, neighbor discovery has also played a role in the security of wireless sensor networks, especially for mitigating control and data traffic attacks. Simple neighbor verification has been found to significantly mitigate the wormhole attack in static sensor networks, [14].

Neighbor discovery is the first step performed by a sensor node upon deployment. Since neighbor discovery requires a very small amount of time, it might be difficult for an adversary to compromise many nodes before the completion of neighbor discovery by the entire network. However, the compromise of even a single node during neighbor discovery can be significantly advantageous to the adversary to attack a variety of existing routing protocols. Also, even external malicious nodes (nodes that do not possess the cryptographic keys) can significantly affect neighbor discovery protocols. They just need to relay packets between two non-neighbors and make them believe that they are neighbors. False neighbor discovery also makes protocols that trust on accurate neighbor discovery, like protocols that fight against wormhole attacks, [14]-[16] and certain routing protocols completely useless.

To understand this, let us consider the following examples. Let there be two legitimate sensor nodes, $S$ and $T$,
which are not within communication range of each other and an adversary $M$ which is within communication range of both $S$ and $T$, as shown in Figure 1.

![Figure 1: A malicious node $M$ fooling two legitimate nodes $S$ and $T$ that they are neighbors. $R(S)$ & $R(T)$ are the transmission ranges of $S$ and $T$ respectively](image)

During neighbor discovery phase, $M$ can fool $S$ and $T$ to believe that they are neighbors by relaying packets between them. After neighbor discovery, since $S$ and $T$ believe that they are neighbors; all communication between them gets controlled by the adversary $M$. If a malicious node $M_1$ colludes with another malicious node, $M_2$, the situation becomes worse. Colluding malicious nodes can make even legitimate nodes that are very far from each other to believe that they are neighbors. This is illustrated in Figure 2. Once a malicious node or a set of colluding malicious nodes make two non-neighbor legitimate nodes to believe that they are neighbors, they can easily create a wormhole and launch a variety of attacks against the data traffic flowing on the wormhole, such as selectively dropping the packets.

![Figure 2: Two malicious nodes $M_1$ & $M_2$ fool two noneighbor nodes $S$ & $T$ to believe that they are neighbors](image)

Therefore, it is necessary for a node to have the capability to verify the claim of neighborhood membership by another node. Research on this topic can be broadly classified into three kinds of approaches to this problem. The first approach assumes that there exists no malicious nodes during the neighbor discovery phase due to which neighbor discovery is always secure and using this assumption, it proposes protocols to prevent other attacks ([14]-[16]). The second approach performs secure neighbor discovery in the absence of wormhole attacks (for example - [18], [19]). Such an approach is obviously not secure since even a single external malicious node can prevent neighbor discovery from being accurate. The final approach takes the wormhole attack into consideration while performing neighbor discovery but it either requires specialized hardware in the form of directional antenna arrays or tight synchronization which might not be feasible for sensor networks ([20], [21]). More importantly, the directional antenna approach does not solve the problem completely.

4 SVNM DESCRIPTION

In many static sensor network applications it is safe to assume initial malicious-free environment during the deployment of the network. However, the situation may change dramatically later. Additional network nodes may be deployed to replace previous malfunction, energy depleted, physically destroyed, or maliciously isolated nodes. We propose here a secure algorithm called SVNM to integrate newly deployed nodes into the existed network.

We propose two versions for secure neighbor verification problem in static wireless networks. By secure neighbor verification, we mean that for any node in the wireless sensor network, no node that is not within its one-hop communication range can become its neighbor.

4.1 SVNM Version I (SVNM_VI)

Assumptions: We assume that a newly deployed node to the existing network cannot be compromised immediately, i.e., before finishing the neighborhood discovery protocol we present below. We also assume that each node knows its current location either through pre-deployment information or using a GPS.

Goal: A newly added node to the network cannot be fooled and integrated into a wrong neighborhood even by a corum of malicious nodes. A node is integrated in a neighborhood when it is accepted as a neighbor by other nodes in that neighborhood.

A node say, $S$, wants to verify the claim by another node say, $T$, that $T$ is within the transmission range, $R$, of $S$. Assume that the transmission power used by legitimate nodes is fixed and equal to $P$. Nodes $S$ and $T$ follow the following algorithm:

1. $T$ includes its location coordinates $(x_T,y_T)$ in a claim and broadcast the claim within its transmission range.
2. $S$ replies to the claim with authenticated message asking $T$ to authenticate its location to $S$. The authentication protocol uses the pre-shared key between $S$ and $T$ [13].
3. $T$ replies to the request of $S$ with a claim authenticated using the shared key between $T$ and $S$. Alternatively, $T$ may authenticate itself to the whole neighborhood using µTesla protocol [22].
4. $S$ adds $T$ to its neighbor list if the location $(x_T,y_T)$ is within the transmission range of $S$.

Security analysis: based on the assumption that $T$ is honest, it will correctly include its current location in the claim. Since the claim is authenticated using the shared key between $S$ and $T$. Malicious nodes $M_1$ and $M_2$ can forge a
different location for $T$. Therefore, $S$ either receives a correctly authenticated claim that includes the correct location or receives an incorrectly authenticated claim. In the first case $S$ can correctly identify whether $T$ is a neighbor or not. In the later case $S$ drops the claim.

4.2 SVNM Version II (SVNM_VII)

**Assumptions:** In this version, we relax the assumption that a node cannot be compromised before completing the neighbor discovery. Moreover, the base station is responsible for updating the whole network using one-hop broadcast about the maliciously detected nodes.

**Goal:** A newly added node to the network cannot integrate into more than one neighborhood and cannot be fooled and integrated into a wrong neighborhood by even a corum of malicious nodes.

1. $T$ includes its location coordinates $(x_T, y_T)$ in a claim and broadcast the claim within its transmission range.
2. $S$ replies to the claim with authenticated message asking $T$ to authenticate its location to $S$. The authentication protocol uses the pre-shared key between $S$ and $T$.
3. Node $T$ replies to the request of $S$ with a claim authenticated using the shared key between $T$ and $S$.
4. Any node, say $J$, that could overhear the authenticated claim of $T$, computes the distance (dist($J, T$)) between its location $(x_J, y_J)$ and $(x_T, y_T)$.
5. Node $J$ sends an authenticated message of accept (if dist($J, T$) \leq R) or reject (if dist($J, T$) > R) to the base station. The message contains the decision of $J$ and the original authenticated claim of $T$.
6. The base station collects the reject and the accept messages from the network. If the number of rejects is greater than a threshold $\gamma$ then the base station adds $T$ to the blacklisted nodes. Else, the base station confirms the location of $T$. If the number of nodes in the new neighborhood of $T$ is less than $\gamma$, then the threshold will be the number of nodes in the neighborhood.
7. The base station sends its decision to the neighborhood of $T$ through a one-hop authenticated message.

5 ANALYSIS

Here we analyze the probability of isolating a node that forges its actual location. We find the probability of isolation as a function of $\gamma$, the number of malicious nodes ($N_m$), the fake transmission range $R_f$, and the node density $d$.

Using similar reasoning as that of SVNM_VI, an honest node cannot be integrated in a wrong location. However, a compromised node can easily fake its location and legitimately authenticate that fake location since it posses the authentication keys. For the latter case, a malicious node may try to falsify the neighbor information alone by lying about its location or increasing its range through high power transmission. On the other hand, a corum of malicious nodes may collaborate to falsify the neighbor information.

In the following analysis, we consider a homogeneous network of $N$ nodes where the nodes are uniformly distributed in the field with density $d$. For simplicity, we assume that the field is large enough that edge effects can be neglected in our analysis. Moreover, we assume the transmission range to be a perfect circle.

In Figure 3, node $T$ extends its range to reach $S$ and fakes its location to be at a distance within the legitimate transmission range ($R$) of $S$. The real location of $T$ is at $(x_r, y_r)$ while its claimed location is $(x_f, y_f)$. Node $S$ will not be able to discover the fake claim of $T$ by itself. However, node $J$ can tell that $T$ is lying since the claimed location makes the distance between $J$ and $T$ greater than the legitimate range $R$. The probability that $J$ detects $T$ is the probability that $J$ lies inside the expanded range of $T$ (the large circle) and outside the claimed range (dotted circle).

![Figure 3: High power transmission](image)

The upper bound on the possible number of nodes that can detect $T$ (such as $J$) occurs when almost the whole claimed range falls out of the expanded range.

$$N_{upper} = \pi \cdot R_f^2 \cdot d - 1$$  \hspace{1cm} (1)

The lower bound on the possible number of nodes that can detect $T$ occurs when the whole claimed range falls inside the expanded range.

$$N_{lower} = \pi \cdot \left(R_f^2 - R^2\right) \cdot d - 2$$  \hspace{1cm} (2)

Assuming reliable communication with the base station, a best case detection occurs when $N_{upper} > \gamma$ and the worst case detection occurs when $N_{lower} > \gamma$. Therefore, the best case and the worst case detection probability of $T$ is given by,

$$P_{det}(best) = \sum_{i=N}^{i=N_{upper}} \left(1 - \frac{N_{upper}}{N}\right)^{i-1} \left(1 - \frac{N_{upper}}{N}\right)^{i-\gamma}$$  \hspace{1cm} (3)

$$P_{det}(worst) = \sum_{i=N}^{i=N_{lower}} \left(1 - \frac{N_{lower}}{N}\right)^{i-1} \left(1 - \frac{N_{lower}}{N}\right)^{i-\gamma}$$  \hspace{1cm} (4)

In general, let the distance between the real location and the fake locations of $T$ be $x$ as shown in Figure 4. Moreover, let $T$ managed to transmit within the fake range of $R_f$ centered at $(x_r, y_r)$ using either high power transmission(e.g.,
Figure 3) or collaboration with other malicious nodes (e.g., Figure 2). The intersection area between the fake and the real locations is given by,

\[
A_{\text{inter}} = R^2 \cos^{-1}\left(\frac{x^2 + R^2 - R_f^2}{2xR}\right) + \\
R_f^2 \cos^{-1}\left(\frac{x^2 + R_f^2 - R^2}{2xR_f}\right) - \frac{1}{2}
\]

\[\sqrt{-(x + R + R_f)(x + R - R_f)(x - R + R_f)(x + R + R_f)}\]

(5)

Therefore, the area in which nodes can detect T is,

\[A_{\text{det}} = \pi R_f^2 - A_{\text{inter}}\]

(6)

Let the number of malicious nodes in the network be \(N_M\) and that they are uniformly distributed in the network. Let the number of good nodes be \(N_G\) \((N_G = N - N_M)\), the probability that a good node is available to detect T is,

\[P_{\text{good}} = \frac{A_{\text{det}} \cdot d}{N} \cdot \frac{N_G}{N}\]

(7)

The probability of detecting T by at least \(\gamma\) nodes and thus isolating it by the base station is given by,

\[P_{\text{isolate}} = \sum_{i=\gamma}^{i=N_G} \binom{N_G}{i} (P_{\text{good}})^i (1 - P_{\text{good}})^{(N_G - i)}\]

(8)

Figure 5 shows the probability of isolation as \(d\) varies from Equation(8). The input parameters are: \(R = 50\), \(R_f = 100\), \(\gamma = 10\), \(N = 100\), \(N_M = 10\), and \(x = 100\). The figure intuitively shows that the probability of isolation increases with \(d\). For the input parameters shown, \(d = 0.0006\) (about 5 nodes within the legitimate transmission range \(R\)) gives probability of isolation of a malicious node trying to fake its location of more than 0.9. With \(d=0.0008\) (6 nodes within \(R\)), we get perfect detection of malicious nodes.

Figure 6 shows the probability of isolation as \(\gamma\) varies from Equation(8) with \(d = 0.001\) and the same input parameters as in Figure 5. The figure shows that we can get perfect isolation of malicious nodes with \(\gamma\) less than 10. Also the figure shows that even with values of \(\gamma\) up to 15 we still get very high isolation probability (more than 0.98).

Figure 7 shows the probability of isolation as \(N_M\) varies from Equation(8) with \(d = 0.001\) and the same input parameters as in Figure 5. The figure shows that even with 40% malicious nodes, we still get isolation probability of more than 0.94.

Figure 8 shows the probability of isolation as \(R_f\) varies from Equation(8) with \(d = 0.001\) and the same input parameters as in Figure 5. The figure shows that with \(R_f = R\), the isolation probability is above 0.75 and increases to perfect isolation with values of \(R_f\) greater than 80.
achieve accurate neighborhood information. We introduce neighbor verification as a necessary tool to achieve accurate neighborhood information in ad hoc wireless networks. We show that neighbor verification is a necessary tool to achieve accurate neighborhood information.

In the first variation (SVNM_VI) of the algorithm, we address the case in which a new node may be compromised before finishing its neighbor discovery. In the second variation (SVNM_VII), we address the general case in which a new node may be compromised before finishing its neighbor discovery. The cost of relaxing the assumption of SVNM_VI in SVNM_VII is the involvement of the base station in the neighbor verification process.

We performed extensive security analysis of SVNM and the results show that we can achieve almost perfect isolation of malicious nodes that lie about their actual location using high power transmission or with the help of other malicious nodes.

6 CONCLUSION

We have shown the importance of having accurate neighborhood information in ad hoc wireless networks. We show that neighbor verification is a necessary tool to achieve accurate neighborhood information. We introduce an algorithm called SVNM with two variations to securely integrate newly added nodes in existing static wireless sensor networks. In the first variation (SVNM_VI) of the algorithm, we address the case in which a new node cannot be compromised before its neighbor discovery. In the second variation (SVNM_VII), we address the general case in which a new node may be compromised before finishing its neighbor discovery. The cost of relaxing the assumption of SVNM_VI in SVNM_VII is the involvement of the base station in the neighbor verification process.

We performed extensive security analysis of SVNM and the results show that we can achieve almost perfect isolation of malicious nodes that lie about their actual location using high power transmission or with the help of other malicious nodes.

7 REFERENCES