Secure Position Verification through Obfuscation

Ming-Yee Iu
Tutor: Maxim Raya

7th February 2005

1 Introduction

In a vehicular network [1], vehicles must be able to determine the positions of other vehicles in the vicinity in order to coordinate their actions. For simple traffic coordination tasks, a radar-based scheme that allows an individual vehicle to determine the relative positions of other nearby vehicles is usually sufficient. But for more complicated circumstances, vehicles need to exchange position information with other vehicles.

By using a technique described by Kuhn [2], it is possible for individual vehicles to calculate their own positions securely using GPS-like systems. Unfortunately, vehicles might not be able to trust position information transmitted by other vehicles. In actuality, this is not a huge problem because driving is inherently unsafe, and having untrustworthy position information does not significantly increase the danger in driving. A crazy driver can easily knock other cars off the road even if other drivers are able to track his or her exact position at all times. Or a driver can also opt to disable wireless transmissions from her vehicle and to alter the radar signature of her vehicle, making herself invisible to other drivers.

Still, a vehicular network with more security than is necessary is better than one with insufficient security, so it is still useful to examine these issues. Obviously, a signature infrastructure needs to be deployed so that identities cannot be forged and so that vehicles cannot retract their transmissions about their reported positions. Then, given that such an infrastructure exists, the main use of secure positioning will be in accident reconstruction, in detecting vehicles with malfunctioning positioning, and in providing location based services. Since vehicles are able to calculate their own positions securely, secure positioning in vehicular networks is mainly a matter of verifying that the reported positions of other vehicles is correct.

This paper examines a scheme for secure position verification using a variant of distance bounding and the use of ”stealth” nodes.

2 Related Work

Currently published robust secure positioning systems generally make use of only one positioning primitive: distance bounding. Distance bounding makes use of fundamen-

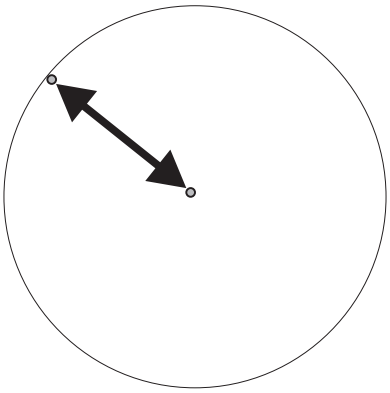

Figure 1: Knowing the distance to a node restricts the possible positions of that node to a circle

tal properties of sound or radiation waves for its security, making it difficult to circumvent.

The general premise behind distance bounding [3] is that if a single node, known as the verifier, wishes to find the distance between it and another node, called the claimant, it can send a ping to the claimant. The claimant, upon receiving the ping, is supposed to immediately reply to the verifier. If all of these communications occur over radio links, then then the signals should travel at the speed of light. The verifier can then calculate the distance between itself and the claimant by measuring the amount of time between its ping transmission and the receipt of a reply, subtracting the amount of delay added by both nodes to the ping reply due to processing, calculating the distance that light can travel in that amount of time, and then dividing that distance by two. Because nothing can travel faster than the speed of light, it is impossible for a claimant to appear closer to the verifier than it actually is (though it can appear farther than it actually is by delaying its reply to the ping). In actual suggested implementations, this general premise is hardened to prevent problems with impersonation, with nodes replying before receiving a ping, or other such attacks.

Such a positioning primitive can be used alone to determine if a node lies within a certain region [4]. Since knowing the distance between a verifier and a claimant restricts the possible positions of the claimant to a circle surrounding the verifier (figure 1), a verifier can determine if a node lies within a certain region if the circle of possible positions lies entirely within the region of interest. To handle non-circular regions, multiple verifier nodes can be used as shown in figure 2.

Distance bounding can also be combined with radar images to associate identities with radar-detected vehicles [5]. By using a radar, a node can detect the positions of nearby vehicles. But the radar can only detect the radar signatures of those vehicles and not their identities. By performing distance bounding on all nearby vehicles, a node can then correspond the measured distances of individual vehicles with the positions of vehicle radar signatures, thereby allowing it to associate an identity with each vehicle. Although this is a very viable and useful scheme, it is limited by the range and possible

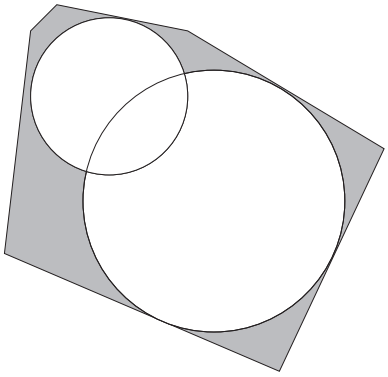

Figure 2: The use of multiple nodes allows nodes to verify whether nodes are within a non-circular region

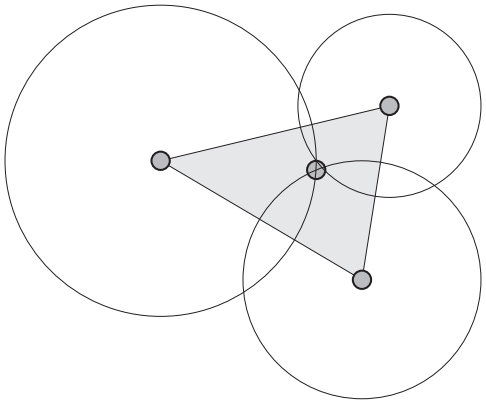

Figure 3: Verifiable multilateration can be used for secure positioning

weaknesses of vehicular radar.

Finally, distance bounding can also be used in a scheme called verifiable multilateration that can use distance bounding measurements from multiple nodes to determine exact node positions [3]. As mentioned before, knowing the distance between a claimant and verifier restricts the possible position of the claimant to a circle around the verifier. In a triangular region formed by three nodes, if all three nodes perform distance bounding on a claimant node within this triangle, the intersection of the three circles defines an exact position for the claimant (figure 3). Since the claimant can only cheat on its distance bounding replies to appear further from a verifier than it actually is, any attempt to cheat will cause the circular regions not to intersect properly, meaning the verifiable multilateration will fail.

It may be possible to build secure positioning systems based on other positioning primitives such as angle of arrival measurement [6] though no such systems have been designed yet.

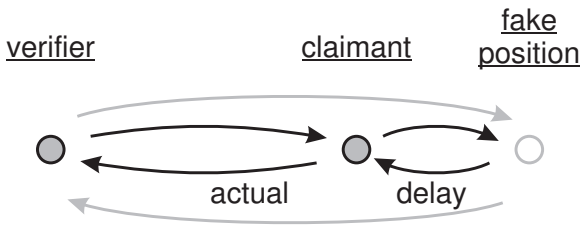

Figure 4: By delaying its distance bound reply, a node can easily appear to be in a different position

3 Building a Secure Position Verification Scheme

This section examines one approach for how one vehicular node can verify the position claims of another vehicular node.

3.1 An Insecure Position Verifier

Distance bounding can be used by a single node to insecurely verify the position of another node. For example, if a verifier node at position A wants to verify that another claimant node is at position B, it can initiate a distance bounding request with the claimant node and find out the distance of the claimant node. Since the distance bounding restricts the claimant to be a certain distance away from the verifier, as long as this distance is consistent with the claimant's reported position, then the claimant's position is verified.

Unfortunately, performing verification this way is insecure because

- Since distance bounding only measures the distance of the claimant from the verifier, the claimant can, in fact, be in any position that is the same distance away from the verifier. As such, the claimant can claim to be in any position that is the same distance away from the verifier as it currently is. This problem can be mitigated somewhat by having several different verifiers perform distance bounding on the claimant.

- A claimant can cheat and respond to a distance bounding request in such a way as to appear further from the verifier than it really is. As such a claimant can claim to be at any arbitrary position whose distance is further from the verifier than it currently is. Since it knows the position of the verifier, it can compute the distance between the verifier and the fake position, subtract its real distance from the verifier, and then delay its response to the distance bound request by twice the time it takes the speed of light to travel that distance. The claimant can get the fake position to be verified as being correct in this way (figure 4).

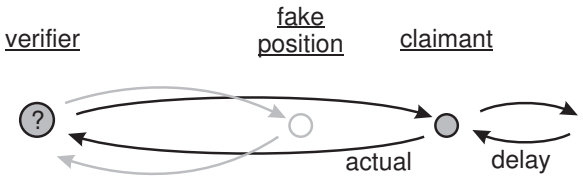

Figure 5: If a claimant does not know the position of the verifier, it risks claiming to be in a position too close to the verifier

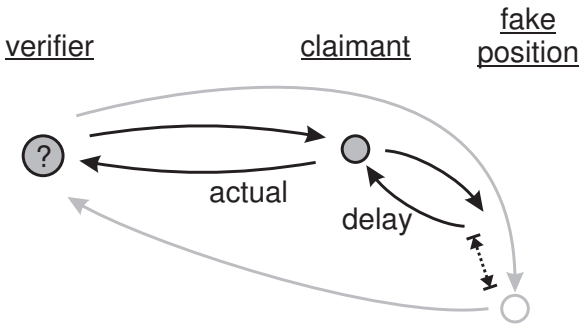

Figure 6: If a claimant does not know the position of the verifier, it does not know how much to delay its distance bound reply so as to be consistent with a fake position

3.2 Securing the Position Verification through Obfuscation

The problems of using a simple verification scheme based on distance bounding disappear, though, if the claimant does not know the position of the verifier. If the claimant always reports its real positions and responds to distance bound requests immediately as expected, then its position will be verified correctly. If it reports a fake position, however, it does not know how to respond to a distance bound request in a way that is consistent with the fake position.

If the fake position happens to be closer to the verifier than the claimant, then the claimant's fake position will not verify correctly because it is impossible for the claimant to respond to a distance bound request to appear closer to the verifier than it really is (figure 5).

On the other hand, if the fake position happens to be further from the verifier than the claimant, the claimant needs to delay its distance bound response in order to be verified correctly. But the claimant needs to know the verifier's position in order to know how long it must delay its response so as to be consistent with its fake position (figure 6).

Unfortunately, it is not too difficult for a claimant to find the position of the verifier. If the verifier is another vehicle, than the other vehicle will periodically report its position to the claimant. Even if this is not the case, the claimant can still use angle-of-arrival measurement, signal strength measurements, and other techniques for estimating the position of the verifier. As such, building a position verification scheme

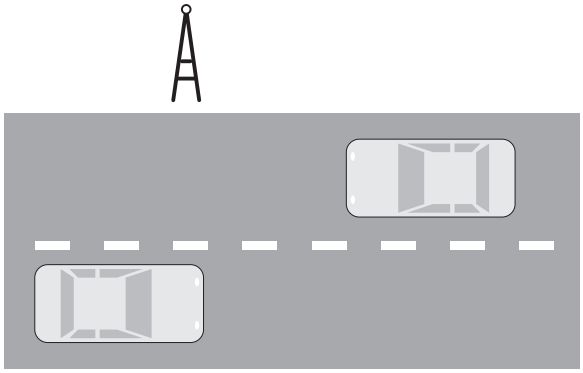

Figure 7: Roadside infrastructure verifies vehicle positions for possible cheaters

upon the idea that the verifier can be hidden is infeasible.

4 Stealth Nodes for Secure Position Verification

Since the verifier must transmit signals and communicate with other nodes, it is not possible to hide its position. It is possible to hide the position of other nodes though, as long as they remain silent and do no transmissions themselves. These nodes, named stealth nodes, can listen in on broadcasted position information and distance-bounding communication and verify reported positions. Although it is possible for a determined cheater to search for these hidden nodes and note their positions, it is difficult to do so in an automated fashion because they do not make any wireless transmissions. If a claimant does not know the position of these stealth nodes, it will be unable to cheat on its position.

The concept of stealth nodes is particularly well suited for performing secure position verification with the help of road-side infrastructure. For example, in a vehicular network such as in figure 7, there might be road-side infrastructure for verifying the positions of vehicles on the road. If there are any vehicles on the road whose positions cannot be verified, the infrastructure will warn all the vehicles in the area to take appropriate action such as possibly slowing down.

One can create such an infrastructure using one active node that initiates distance bounding challenges to passing vehicles and a passive stealth node that monitors distance bounding communications and checks for inconsistencies. The passive node needs to be connected with a wire to the active node so that the passive node will have the necessary keys from the active node to decrypt communications involving the active node (if necessary), and so that the passive node can send warnings about failed position verifications to the active node without revealing its position to others.

During a distance bounding exchange between an active node and a vehicle, the active node will send a ping to the vehicle and the vehicle will reply. A passive node that overhears the ping and the reply can measure the time difference of arrival (TDOA) [7] between the two messages. Since the passive node knows its own position and the

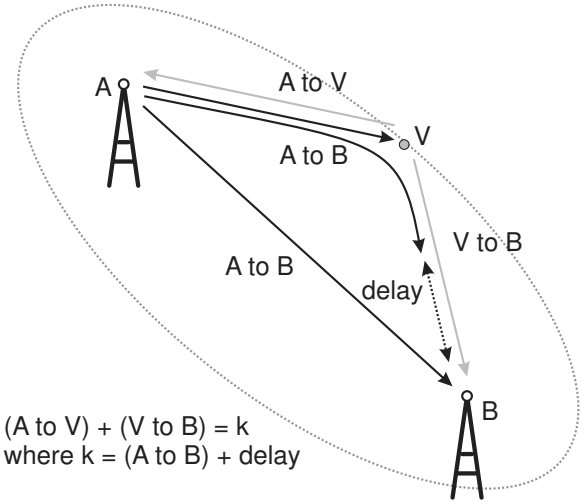

Figure 8: Since B knows A's position and the delay between when it received A's signal and V's reply, it can restrict V's position to be on an ellipse

position of the active node, it can use the time difference to restrict the position of the claimant to an ellipse with foci at the active and passive nodes (whereas normal distance bounding restricts the claimant to a circle centred on the verifier). Figure 8 shows how the formula for the ellipse is calculated. If the self-reported position of the claimant is not within a certain error distance of the ellipse, the passive node can signal this detected inconsistency to the active node, which will then send out a warning to all other vehicles. Because the passive node is always silent, claimants will have difficulty finding its position. If they do not know the passive node's position, they do not know the positions of one of the foci of the ellipse that they must pretend to be on, and hence cannot cheat on their reported positions and distance bounding responses in a systematic way. Since this positioning scheme relies on the hiding of stealth nodes for its security, it is essentially obfuscation-based.

5 Simulation

The security of secure position verification through obfuscation lies in the ability to hide the positions of stealth nodes and in the difficulty of a vehicle to know how to give false but consistent answers to a distance bounding challenge. It is still possible, however, for a vehicle to stumble upon fake positions and consistent distance bounding responses randomly. The probability that a vehicle will be able to generate such combinations are dependent on the position of the vehicle and layout of the highway, but these scenarios can be examined by simulation.

In this paper, we simulate a scenario involving a straight 1 km stretch of a two-lane highway (figure 9). The highway has two shoulders of 2.4 m each and two lanes of 3.6 m each, resulting in a total width of 6 m [8]. Street lights are placed at an

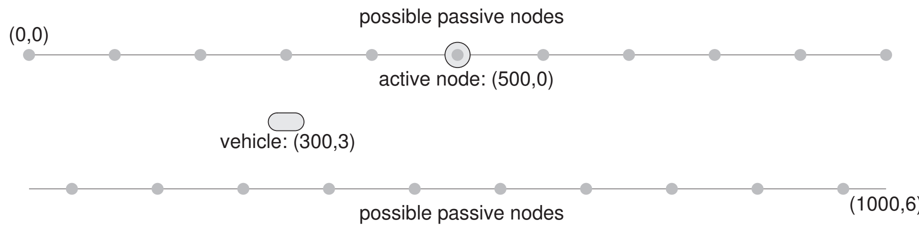

Figure 9: Positions of nodes in the base simulation scenario

71	71	71	71	71	71	100	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	95	28	28	28	28	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	95	28	28	28	28	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	95	28	28	28	28	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	100	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	100	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	100	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	100	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	100	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	95	28	28	28	33	33	33	33	33	33	33	33	33	33	33
71	71	71	71	71	71	95	28	28	28	33	33	33	33	33	33	33	33	33	33	33

Figure 10: Base Scenario Simulation Results

arbitrary interval of 50 m on alternating sides of the highway. It is assumed that wireless infrastructure nodes must be installed on the street lights. An active infrastructure node is installed on the traffic light in the middle of the stretch of highway. A car is placed on the median of the highway 200 m from the active node. When responding to a distance bounding request, the sum of the distance between the car's reported position and the active node and the distance between the car's reported position and the passive node must be within an arbitrarily chosen error distance of 2 m from what distance bounding implies that the sum should be.

Given that the car knows the position of the active node and knows that the passive node is installed at a street light somewhere on the 1 km stretch of road, the simulator takes different positions along the highway and calculates the distance bound response from the car which will have the highest probability of being consistent with the given position. It calculates this distance bound response by looking at all distance bound response delays of between 0 seconds and 3.5 microseconds in nanosecond increments. If the passive node is assumed to have a uniform probability of being installed at any street light on the 1 km stretch of road, the probability that a given distance bound response is consistent with the reported position is the number of passive node positions for which the distance bound response is consistent divided by the total number of possible passive node positions.

Figure 10 shows the results of the simulation. The numbers show the percentage probability that a vehicle can successfully be verified as being in a certain position provided that the vehicle uses the best cheating strategy available to it. The scale is not the same vertically and horizontally. There is 1 m between each vertical probability and 50 m between each horizontal probability. Obviously, the vehicle has a 100% probability

42	42	42	42	42	42	42	42	42	42	42	38	100	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	38	38	95	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	38	38	95	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	38	38	95	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	38	100	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	38	100	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	38	100	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	38	100	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	38	100	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	38	95	61	61	61	61	61	61	61	61
42	42	42	42	42	42	42	42	42	42	42	42	95	61	61	61	61	61	61	61	61

Figure 11: Base Scenario Simulation Results with Vehicle Moved Closer to and to Opposite Side of Active Node

71	71	71	71	71	71	100	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	95	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	95	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	95	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	100	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	100	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	100	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	100	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	100	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	95	0	0	0	0	0	0	0	4	4	4	4	4	4	4
71	71	71	71	71	71	95	0	0	0	0	0	0	0	4	4	4	4	4	4	4

Figure 12: Base Scenario Simulation Results with Active Node Also Checking for Distance Bounding Inconsistencies

of being verified as being at its real position. With 2 m allowable error on distance bounding, the vehicle can fairly easily pretend to be in nearby positions. Because a vehicle can only delay distance bound responses, it is difficult for it to appear closer to the active node than it actually is. This also makes the results non-symmetric, so that it has difficulty pretending to be on the opposite side of the active node than it really is. Moving the vehicle to the other side of the active node, as in figure 11, shows that the non-symmetry does flip as expected. Unfortunately, it is fairly easily for a vehicle to pretend to be in a position that is further from the active node than it actually is provided that the position is on the same side of the active node as it actually is. This occurs because most of the potential passive node positions are hundreds of meters away, and all of these positions are approximately in the same direction from the vehicle. As such, delaying a distance bound reply by the same amount can simultaneously satisfy the verification performed by nodes from all of those potential positions simultaneously.

In the simulation, there are two infrastructure nodes: an active node and a passive node. So far, only the passive node has been used for verification, but the active node can also check distance bounding responses for consistency as well. Figure 12 shows the effect of enabling verification by both the active node and passive node. Although the results do improve somewhat, a vehicle still has a high probability of being able to successfully pretend to be further from the active node than it actually is.

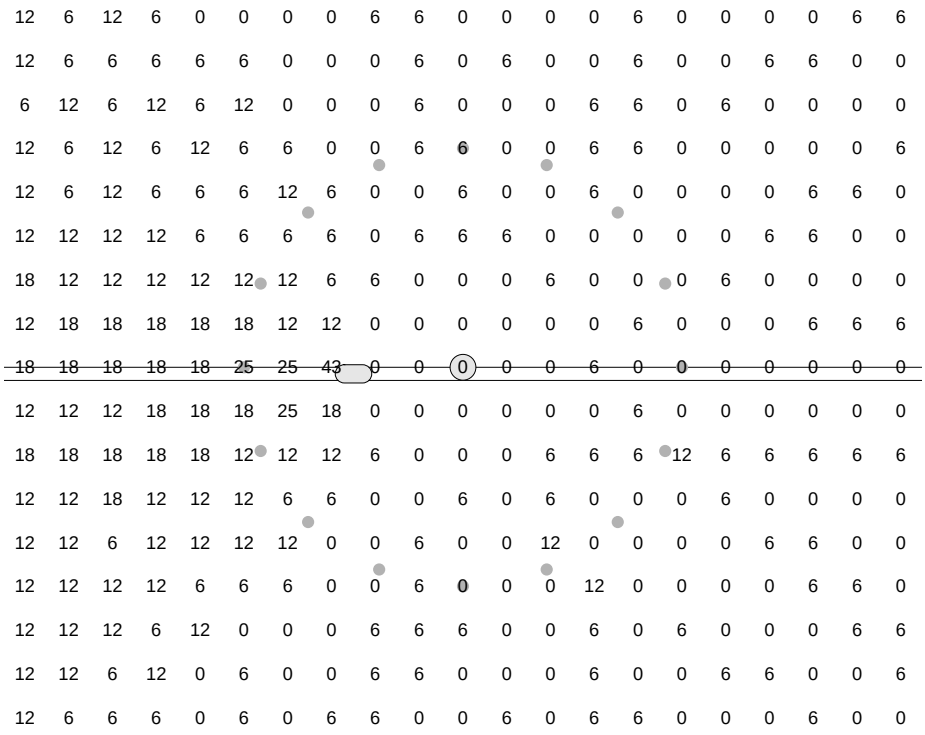

Figure 13: Simulation Results with Passive Nodes Arranged in a Circle

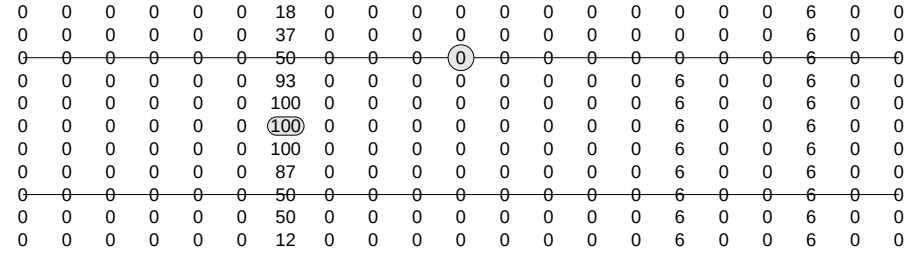

Figure 14: Simulation Results with Passive Nodes at $(67 + n \cdot 134, 500)$ for $n \in [0, 7]$ and at $(n \cdot 134, -500)$ for $n \in [0, 7]$

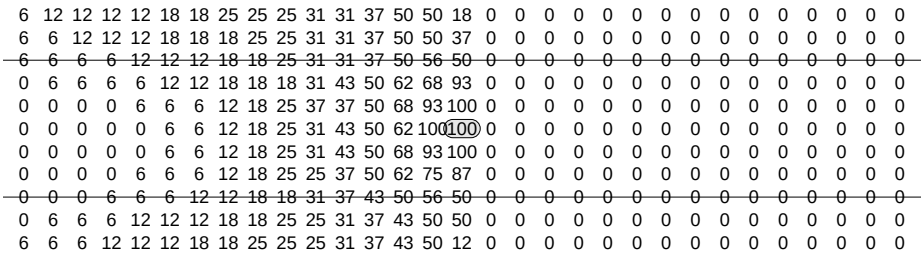

Figure 15: Same Simulation Results as in figure 14 But in Greater Detail

Since the problem with the scenarios so far has been that many of the potential passive node positions are at a similar angle to the vehicle, one arrangement that can avoid this problem is a circular arrangement of passive nodes. Figure 13 shows a scenario where passive nodes can be placed in any of 16 positions, which are each the same distance from the active node and equally spaced angle-wise. The vehicle is still in the middle of the road, but is only 50 m from the active node. The distance between horizontal and vertical probability values is 20 m, and both the active and passive node perform distance bounding checks.

In actuality, it is also possible to ensure that passive nodes are at varied angles from the vehicle by simply pushing the passive nodes further back from the road. For example, figure 14 shows 16 nodes arranged at approximately 67 m intervals along the road, except they alternate at being 500 m off from the left side of the road and at being 500 m off from the right side of the road. Again, both the active node and passive node participate in distance bounding. The distance between horizontal probabilities is 50 m and the distance between vertical probabilities is 1 m. The results seem significantly better than before.

Figure 15 shows the above diagram in greater detail. The diagram is centred over the vehicle, with the distance between vertical probabilities being 1 m and the distance between horizontal probabilities being 2 m. It seems that a vehicle still has some ability to cheat by a small distance in its horizontal position.

It's possible to trade off security along the horizontal axis for more security along

6	12	12	12	12	12	12	12	18	18	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	12	12	12	12	12	12	18	18	25	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	12	12	12	12	12	12	18	18	25	25	37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	12	12	12	12	12	12	18	18	25	25	37	56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	12	12	12	12	12	18	18	25	25	31	56	81	100	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	12	12	12	18	18	18	25	31	37	50	62	75	100	100	68	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	12	12	12	18	18	25	31	37	43	50	56	68	100	100	93	37	0	0	0	0	0	0	0	0	0	0	0	0
12	12	18	18	18	18	25	31	37	37	43	50	50	50	56	100	93	37	31	0	0	0	0	0	0	0	0	0	0	0
12	12	18	18	25	25	31	31	37	37	43	37	37	31	12	0	6	25	31	25	25	0	0	0	0	0	0	0	0	0
12	18	18	25	25	25	31	31	31	31	25	18	6	0	0	0	0	6	18	25	25	18	0	0	0	0	0	0	0	0
18	18	25	25	25	25	31	31	25	18	12	0	0	0	0	0	0	0	6	12	18	18	18	0	0	0	0	0	0	0

Figure 16: Simulation Results with Active Node at $(500, -500)$ and passive nodes at $(n \cdot 67, 500)$ for $n \in [0, 15]$

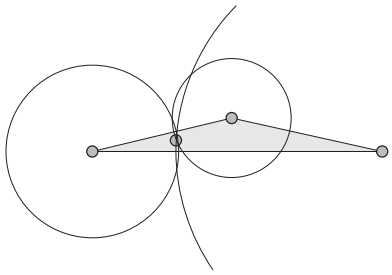

Figure 17: When the triangular region formed by three verifier nodes is too thin in verifiable multilateration, the circle tangents end up orienting the same way at the intersection point, decreasing resolution along that axis

the vertical axis. By putting the active node 500 m from the road, and have the possible positions of the passive nodes be 500 m from the other side of the road, it improves the vertical sensitivity of the distance bounding. Figure 16 shows this scenario, with the distance between vertical probabilities being 1 m and the distance between horizontal probabilities being 2 m.

6 Verifiable Multilateration using TDOA Techniques

It may be possible to improve the security of the system described so far by having two stealth nodes instead of one. However, with three nodes, it becomes possible to build a secure position verification scheme based on verifiable multilateration, which does not require nodes to be connected by wires or rely on keeping the positions of nodes secret for its security.

Of course, with three nodes, it may be feasible to build a secure positioning system using TDOA for its multilateration that does not rely on secret node positions for its security, does not require nodes to be connected by wired links, and offers better performance than standard verifiable multilateration.

The main weaknesses of a secure positioning system based on normal verifiable

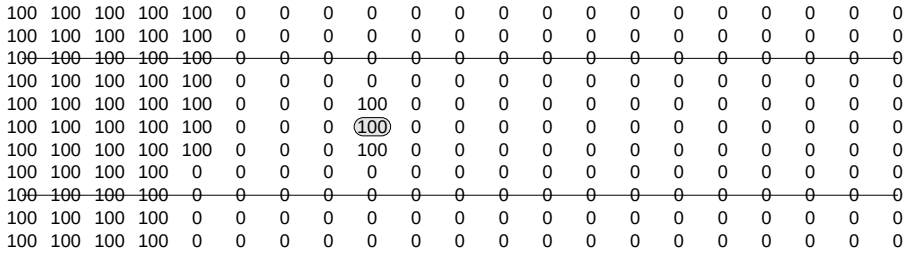

Figure 18: Verifiable Multilateration, using nodes at $(500, 100)$, $(300, 100)$, and $(700, 100)$, with Vehicle at $(400, 3)$ Inside Triangular Region

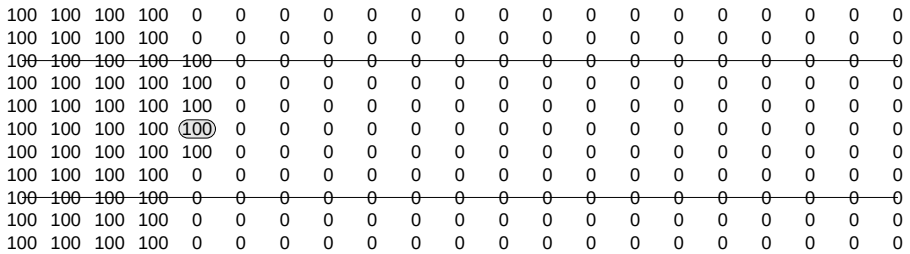

Figure 19: Verifiable Multilateration, using nodes at $(500, 100)$, $(300, 100)$, and $(700, 100)$, with Vehicle at $(200, 3)$ Outside Triangular Region

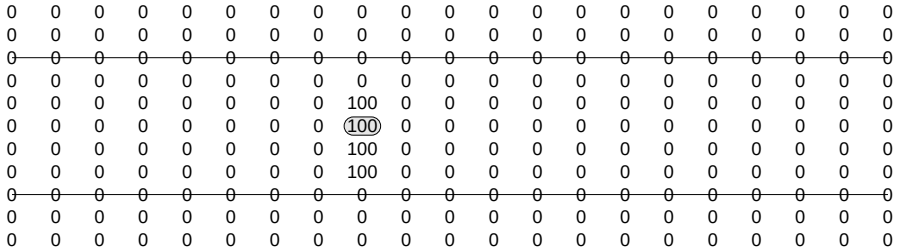

Figure 20: TDOA Multilateration with Vehicle Inside Triangular Region

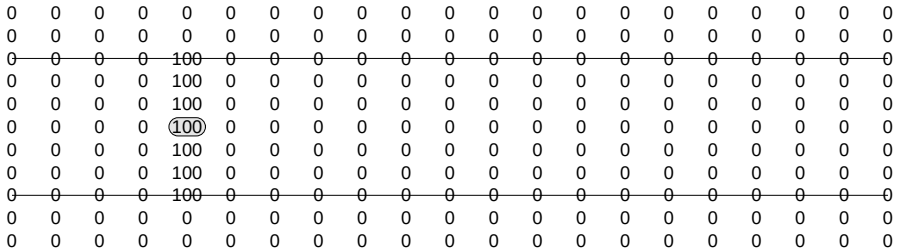

Figure 21: TDOA Multilateration with Vehicle Outside Triangular Region

multilateration is that it requires three distance bounding communications and that it requires vehicles to be within a triangular region formed by three nodes for it to work properly. If a node is outside of this region, the nodes cannot accurately determine the position of the vehicle. As such, often many nodes are needed in order to ensure a good coverage of triangles over a road surface. Also, these nodes have to be placed far from the road in order to provide a good ability at detecting vertical movement because otherwise the resulting triangle formed by the nodes is too thin (figure 17). Figures 18 and 19 demonstrate how a vehicle has much more ease in pretending to be in a different position when it is outside of the secure triangular region than when it is inside. In both diagrams, the distance between vertical probabilities is 1 m and the distance between horizontal probabilities is 50 m. One node is placed 100 m from the road, while two other nodes are placed 200 m on either side of that first node, but 100 m from the other side of the road. In figure 18, the vehicle is in the centre of the road, 100 m from the horizontal position of the active node, while in figure 19, the vehicle is 300 m from the horizontal position of the active node.

In a multilateration scheme that uses TDOA, a single node can initiate a distance bounding request with a vehicle, and then the two other nodes can verify the distance bounding by listening to the time difference between when the ping and the reply arrives to them. Since the security of the system does not rely on the positions of the nodes being secret, the nodes can communicate with each other wireless and hence do not need to be connected by wired links. Figures 20 and 21 demonstrate the same scenario as shown before for standard verifiable multilateration, but with the centre

node doing the standard distance bounding, and the other nodes simply verifying the distance bounding. Notice that even when the vehicle is outside the triangular region formed by the three nodes, the vehicle is still fairly limited in its ability to pretend to be in another position.

One potential problem with this approach is that a vehicle with a directional antenna can give different distance bounding replies to each node, meaning that under such an attack, TDOA multilateration is likely no better than normal verifiable multilateration.

7 Conclusion

This paper examines one possible way that obfuscation can be used to build a secure position verification system. Simulation results show that the scheme is reasonably effective provided that infrastructure nodes are carefully placed. The optimal placement of such nodes is to have a single active node placed far from the side of the road, and to place the passive node somewhere far from the road on the other side of the road.

Unfortunately, although this scheme makes use of only two nodes, the two nodes must be joined by a wired link. Also, since the nodes have to be far from the road, there could be problems with supplying power to the nodes, servicing the nodes, obtaining right-of-way, or with objects obstructing radio signals travelling between the road and the nodes. Of course, alternate schemes based on verifiable multilateration requires three nodes and these nodes also have to be placed a reasonable distance away from the road, so they suffer from similar problems.

Although basing any security scheme on obfuscation is usually a bad idea, the secure positioning scheme described here does provide the advantage of being slightly more resistant to attacks involving collusion than other schemes since colluders will need hidden missing information needed to launch an attack. If the system has a large number of stealth nodes as well, then even if the locations of stealth nodes are discovered, the system can still offer a level of security similar to that of normal verifiable multilateration. As such, secure position verification through obfuscation provides a possible alternative to multilateration-based schemes for secure positioning.

In the future, this concept of secure position verification through obfuscation should be explored further. The possibility of using angle-of-arrival measurement and mobile stealth nodes should be examined. The results should be extended to 3-d and more detailed comparisons with other positioning techniques should be undertaken. If the concept compares favourably, test implementations should be made.

References

[1] Broady Cash, Lee Armstrong, and James Arnold. 5.9 ghz dedicated short range communication (dsrc) overview. *http://www.leearmstrong.com/DSRC Home/General Info/DSRC General What is.htm*.

[2] Markus G. Kuhn. An assymetric security mechanism for navigation signals. In *Information Hiding 2004*, pages 239–252.

[3] Srdjan Capkun and Jean-Pierre Hubaux. Secure positioning of wireless devices with application to sensor networks. In *to appear in Infocom 2005*.

[4] Naveen Sastry, Umesh Shankar, and David Wagner. Secure verification of location claims. In *WiSe '03: Proceedings of the 2003 ACM workshop on Wireless security*, pages 1–10. ACM Press, 2003.

[5] Srdjan Capkun and Jean-Pierre Hubaux. Sivp: Secure radar-aided inter-vehicular positioning. In *not yet published*.

[6] Dragoş Niculescu and Badri Nath. Vor base stations for indoor 802.11 positioning. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 58–69. ACM Press, 2004.

[7] I. Jami, M. Ali, and R.F. Ormondroyd. Comparison of methods of locating and tracking cellular mobiles. In *IEE Colloquium on Novel Methods of Location and Tracking of Cellular Mobiles and Their System Applications*, 1999.

[8] California Department of Transportation. *Caltrans Highway Design Manual*, chapter 300: Geometric Cross Section. Caltrans Publication Unit, 2004.