

Privacy in Inter-Vehicular Networks: Why simple pseudonym change is not enough

Björn Wiedersheim, Zhendong Ma
Institute of Media Informatics
Ulm University, Germany
givenname.surname@uni-ulm.de

Frank Kargl
University of Twente,
The Netherlands
f.kargl@utwente.nl

Panos Papadimitratos
Laboratory for Computer
Communications and Applications
EPFL, Switzerland
panos.papadimitratos@epfl.ch

Abstract—Inter-vehicle communication (IVC) systems disclose rich location information about vehicles. State-of-the-art security architectures are aware of the problem and provide privacy enhancing mechanisms, notably pseudonymous authentication. However, the granularity and the amount of location information IVC protocols divulge, enable an adversary that eavesdrops all traffic throughout an area, to reconstruct long traces of the whereabouts of the majority of vehicles within the same area. Our analysis in this paper confirms the existence of this kind of threat. As a result, it is questionable if strong location privacy is achievable in IVC systems against a powerful adversary.

I. INTRODUCTION

Inter-vehicle communication (IVC) systems have been actively researched over the past years. Vehicles that can communicate with each other and road-side units (RSUs) enable a range of applications. For example, applications that provide warnings on road dangers and traffic jams, or those that offer comfort enhancements (e.g., automated update of point-of-interest information to car navigation systems). Many of the envisioned IVC protocols and applications rely on position and time information. This requires all vehicles frequently broadcasting their position, combined with a time stamp of the message generation, openly to all of its neighbors.

As vehicle transmissions can be eavesdropped by anyone within radio range, there exists a clear threat: location information could be collected and misused [18]. By establishing a network of RSUs, any public, private, commercial, or criminal attacker can collect these packets and create detailed location profiles of vehicles and consequently their drivers. Possession of such location profiles could easily breach the privacy of drivers, as there is usually a strong correlation between a vehicle and its driver; most vehicles are used by only very few drivers [8].

IVC protocols and applications provide various identifiers of the vehicle, in particular the vehicular communication equipment. This can be an identifier for a networking protocol or an identifier for an application. We abstract away implementation details and consider the basic problem at hand: the correlation of an identifier ID with a time t and a location l . The (ID, t, l) tuple is called a *location sample*, and a location profile is set of multiple tuples (ID, t_i, l_i) for the same identifier ID , with i simply the index of sample.

In order to enhance privacy, one could blur the information such a profile provides. For example, by decreasing the

accuracy of the data in a location sample, that is, the time information, t , or the location information, l . Providing inaccurate location information is also no option, as many VANET applications require very high location accuracy to determine the correct position and lane of neighboring vehicles. Location obfuscation, i.e. the intentional blurring of the own position, would render such applications useless. Moreover, the attacker receiving directly messages sent from a vehicle directly can easily record a time stamp t on its own.

A feasible approach is to hide the ID of the sending vehicle from eavesdroppers. Nonetheless, most communication and application protocols require a unique identifier, for example as source or destination address of packets. The middle-ground solution for such protocols is to use a pseudonym, $PSNYM$, instead of the ID . Pseudonyms do not contain any identifying information, e.g., no vehicle identification numbers (VIN), clearly no driver names, and it simply identifies the vehicular node. We term a tuple (t_i, l_i) an *anonymous position sample*, and a $(PSNYM, t_i, l_i)$ tuple a *pseudonymous position sample*.

Still, pseudonymous position samples can be collected and combined into *pseudonymous location profiles* $(PSNYM, t_i, l_i)$. An attacker that manages to obtain such pseudonymous location profiles could relatively easily relate them to specific vehicles; off-line information, could be obtained via cameras, and profiles could be correlated to specific areas (e.g., profiles starting/ending on weekday mornings at the same location would likely reveal home and work addresses that could then be connected to individuals).

Therefore, the use of a single pseudonym is not enough to protect privacy. To address this problem, solutions in the literature propose that each vehicle use multiple pseudonyms, changing frequently from one pseudonym to another [16]. The attacker could then only record location profiles, also denoted in the rest of the paper as *tracks*, each of them consisting of tuples of the form $(PSNYM_x, t_i, l_i)$ with each $PSNYM_x$ representing one of the pseudonyms used by a node. Use of changing pseudonyms can be considered the state-of-the-art in VANET privacy enhancing technologies; such schemes were designed with the intention to thwart adversaries that eavesdrop parts of the network.

However, the accurate location and time information IVC messages contain, and the very frequent transmission of

messages (typically, for transportation safety, 10 messages per second) raise an important question. Can pseudonymous location samples with different pseudonyms be linked to each other? Or, even further, could anonymous location samples be linked? If so, lengthy over time tracks of vehicles could be generated by an adversary. If the adversary covered large areas and obtained rich sets of location samples, it could then create extensive location profiles.

In this paper, we investigate this question. We consider an area where that the adversary can collect IVC messages, notably pseudonymous location samples. We analyze how effectively it can create location profiles, that is, essentially, for how long it can extract tracks for the same vehicle. Utilizing one approach relating to the problem of multi-target tracking, in particular Multi-Hypothesis-Tracking (MHT) [20], we find that linking between samples under different pseudonyms for the same vehicle can be surprisingly successful under various system setups. This clearly indicates that the adversary considered here can indeed significantly weaken the location privacy of vehicular network users. Moreover, it points out the need of additional investigations on the location privacy that can be achieved given the constraints of vehicular communication systems.

In the rest of the paper, we first review the related work in Sec. II and outline the system and adversary models (Sec. III). Then, we describe our tracking approach in Sec. IV and present the results of our simulation-based study in Sec. V; these results demonstrate the effectiveness of our approach. Sec. VI summarizes lessons learned.

II. RELATED WORK

Changing pseudonyms is regarded as one of the best solutions so far to the privacy problem in vehicular communications. Many approaches have been proposed to address different aspects in the pseudonym life cycle. [17] proposed a pseudonym-based security architecture which covers the management and organizational issues of pseudonyms. It also introduced a framework on how to change pseudonyms. [4] devised mechanisms to improve the efficiency and robustness of pseudonym generation and authentication in VANETs. [19] suggested to change pseudonyms according to the vehicle speed. [6] focused on integration and implementation of pseudonymity support for a realistic VANET communication stack. [23] found that frequent pseudonym change can have a negative impact on communication performance. The lack of an omnipresent infrastructure and loose connectivity in vehicular networks pose challenges on propagating information on revoked pseudonyms; [13] proposes multiple mechanisms to revoke misbehaving and faulty nodes.

Although progresses have been made in pseudonym generation, management, and application, there are few studies on the effectiveness of pseudonym changes in terms of achieved privacy levels and how to maximize protection of location privacy. [12] applies the concept of mix-zones first introduced in [2] and studies of the effectiveness of changing pseudonyms:

the unobservable by the adversary regions are modeled as mix-zones, and essentially nodes change pseudonyms when they traverse such a zone. They find that an adversary can still successfully track vehicles, if it places receivers at half of the intersections in the road network. [7] proposed a context-aware pseudonym change algorithm to improve pseudonym change effectiveness and identified parameters and potential tracking algorithms that influence the attacker's success rate. [21], [10] showed that an attacker can use correlation tracking to link changing pseudonyms by assigning a non-uniform probability distribution to the target anonymity set and choosing the target with the highest probability.

A more recent work on measuring the effectiveness of pseudonym systems in vehicular communication systems is presented in [14], [15]. The authors propose a comprehensive privacy metric to assess the anonymity level provided by pseudonym systems. The results of our work presented here can be used to determine the source-destination trip probability required for this metric.

[9] related the issue of linking anonymous location samples to the data association problem in target tracking systems. Their experiment applies Reid's algorithm for Multiple Hypothesis Tracking (MHT) [20] to track anonymous Global Positioning System (GPS) data, generated by a group of students in and around a university campus. Anonymous location samples from three different tracks are used as input to the MHT algorithm. The tracking results show that despite several temporary incorrect assignments, most anonymous samples can be associated with the correct tracks. This demonstrated that anonymous location samples can be linked by a tracking algorithm such as MHT to reveal user movements, even though their experiment was of limited scope. Inspired by [9], we applied the MHT approach to vehicular networks. We take a similar approach, applying a multi target tracking algorithm on anonymous location samples, but in a more complex and larger scale setting that has different characteristics (VANET vs. pedestrian).

III. SYSTEM AND ADVERSARY MODEL

We assume that vehicles participating in the vehicular network send beacon messages at regular intervals. Those beacons carry only an identifier and the vehicle's current position. To protect from simple location tracking, the vehicles use pseudonymous identifiers and change their pseudonym regularly. In the best case for privacy protection, a new pseudonym is used for each packet sent. We note that we do not dwell on the details of pseudonym construction and the exact node identification; e.g., the pseudonyms can be elliptic curve public keys and whenever there is pseudonym change there is also a change of node identifiers (e.g., network addresses) [16]

In terms of the adversary, we assume a passive attacker with perfect eavesdropping capabilities, i.e. our attacker can receive all beacon messages sent in the network. One might object that such an attacker is quite unlikely to occur, as it is hard to cover a large area with an eavesdropping infrastructure.

However, this attacker model allows us to determine the effect of a “perfect attacker” and later serve as a benchmark for future results for less powerful attacker models.

Moreover, such a global adversary might not be as unlikely as it initially seems; at least within a given area the adversary could have complete coverage. One example is a widespread distribution of micro roadside units deployed during road construction. Or, advances in the design of smart directional antenna arrays might allow large coverage areas with only limited number of adversarial nodes or reliable reception in harsh conditions (congested medium). Or the attacker could tap on other wireless infrastructure.

If such infrastructures are used to create large collections of VANET-related information, notably location and time, such data could be abused to mount exactly the kind of attacks we consider here. After having collected a large quantity of anonymous position samples ($PSNYM_x, t_i, l_i$), such an attacker would use the MHT algorithm to connect those samples to anonymous location profiles. As motivated earlier, connecting such anonymous location profiles to real identifiers is an easy final step [8].

Our goal is to determine the harm that can be done to a driver’s privacy in a vehicular network. We assume the prevalent model for such systems and security architectures, and a powerful global adversary. Our intention is to consider in a sense a worst-case scenario that is not nonetheless very far from what a motivated, strong adversary could achieve. A partially present adversary or an adversary that does not receive all location samples would be clearly weaker, and in principle any given privacy enhancing scheme would achieve better protection. On the other hand, here we experiment and evaluate the tracking abilities of the adversary equipped with a specific mechanism. Accordingly, additional knowledge could be available and more advanced techniques could be devised. All such aspects are parts of future work.

IV. TRACKING APPROACH

A. Multi-Hypothesis Tracking and Kalman filtering

Multiple-target tracking has applications in many areas, such as aircraft tracking, surveillance, and visual tracking. Generally speaking, multiple targets are moving in a given area and their positions are sampled, at random or periodic intervals. Then, the tracking algorithm associates position measurements of the targets in order to form appropriate tracks, that is, a sequence of measurements deemed to belong to (be related to) the same target. In other words, a track reveals the target’s movement in space and time. The difficulty lies in generating correct associations in spite of the noise and errors that usually accompany position measurements. This is known, in the target tracking community, as the data association problem.

Multiple Hypothesis Tracking (MHT) is an established algorithm in the field. It addresses the data association problem, by generating a set of data association hypotheses each time a new batch of measurements arrives. Each hypothesis is a possible association of a measurement with a target. Then,

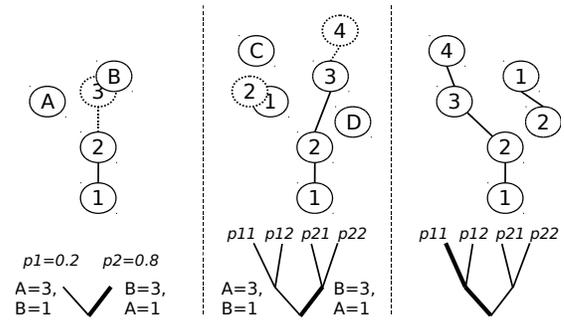


Fig. 1. An example to illustrate MHT

the probability for each of the hypotheses to be correct are calculated, and the hypothesis with the highest probability is chosen as the best solution. MHT relies on Kalman filters [11] to estimate the state variables of each target. In the case of tracking vehicles on a two-dimensional plane, the state variables are the positions and velocities of the targets.

First, we use a simple example to illustrate how the MHT implementation works. Consider the situation in Fig. 1: in the left panel a target has a previously determined track of (1, 2). The MHT uses the Kalman filter to estimate the next position (3) based on the earlier results. Then, the MHT receives two measurements, A and B, which can be either associated with the existing track or be used as the origins of new tracks. In our case, the MHT generates two hypotheses (illustrated in the form of a tree at the bottom) to account for the possible associations.

The two alternatives are annotated with probabilities. As B is closer to the estimated position (3) than A, the probability of hypothesis (1, 2, B) is higher than that of (1, 2, A). Therefore, the analysis suggests that the most likely associations are $B = 3$ and that A is the first point of a new track. Next, the MHT uses the Kalman filter again to estimate a new set of positions for each of the two hypotheses.

The middle panel in the figure shows that the Kalman filter estimates two new positions (2) and (4) based on the hypothesis with $B = 3$ and $A = 1$. The other alternative ($A = 3$ and $B = 1$ in the first step) is not illustrated. After the MHT receives two new measurements, C and D, new hypotheses are generated as branches of the previous ones. At this point, the measurements C and D can be associated either with (2) or (4).¹ Based on the distance of each measurement to the estimated position, the MHT calculates the probabilities for each hypothesis. It is obvious that D is neither close to (4) nor close to (2), therefore p_{21} and p_{22} will be very small. As a consequence, the hypothesis p_{11} illustrated in the right panel is eventually chosen as the most likely one, as it yields the highest probability when both steps are considered.

Next, we provide a more formal introduction of the MHT algorithm and the Kalman filter. Interested readers are referred to the original paper [20] and other literature such as [3] for detailed descriptions. In its simplest form, the process

¹Corresponding to p_{21} and p_{22} ; the alternatives p_{11} and p_{12} are not shown.

of the MHT can be divided into three steps. Those steps run repeatedly for each discrete time step. When receiving new measurements, the MHT first uses the Kalman filter to predict the state variables for each target. Second, hypotheses for each measurement are generated, and their probabilities are calculated based on the deviation between prediction and measurement. The hypotheses are used to build a hypothesis tree. Third, for each new hypothesis, parameters of the Kalman filters are updated with the new measurements. To prevent a state explosion, various reduction techniques are applied, as detailed in Sec. IV-B.

The following equations give an overview of the three steps. The state of a target x at time k is modeled as a linear equation:

$$x_k = Ax_{k-1} + w \quad (1)$$

where x_{k-1} is the state of x in the previous time step, A is called state transition matrix, and w is the disturbance noise. The state variables are building a vector of the form:

$$x = \begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \end{bmatrix}$$

where (p_x, p_y) and (v_x, v_y) are x 's position and velocity, respectively, in an x - y plane. The state variables can be related to a measurement z as:

$$z_k = Hx_k + v \quad (2)$$

where H is a measurement matrix and v is the measurement noise. The noises in the above two equations, w and v , are assumed to be white, with normal distribution with zero means and covariances Q and R , respectively. The matrices Q , R , A and H are of the form:

$$Q = \begin{bmatrix} qT & 0 \\ 0 & qT \end{bmatrix}, \quad R = \begin{bmatrix} r & 0 \\ 0 & r \end{bmatrix},$$

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

where A is used to estimate the state variables and the error covariance from the previous time-step, $k-1$, to the state at the current step, k , and H is used to relate the state x_k to the measurement z_k . These matrices can be adapted over time if detailed information about movement patterns or the measurement process is available.

As we are using two-dimensional position information for our tracking, we chose a generic, time-invariant form for A and H , which was also proposed in [20]. Q is determined by T , which is the difference between the time of the actual prediction and the time of the previous measurement, and a value q . R is only determined by a value r . These two parameters had to be set very carefully, as they have a strong influence on the results of the Kalman filter. The values that we used for our tracking setup are described in Sec. V-A.

When the algorithm receives a new set of measurements at time k , it uses the Kalman filter to estimate the state of a target \bar{x}_k and its error covariance \bar{P} caused by the disturbance noise:

$$\begin{aligned} \bar{x}_k &= Ax_{k-1} \\ \bar{P}_k &= A\bar{P}_{k-1}A^T + Q \end{aligned} \quad (3)$$

where \bar{P}_{k-1} is the error covariance at the previous time-step, $k-1$.

Based on the set of new measurements Z^k , the algorithm generates a set of hypotheses Ω^k . Each hypothesis represents a different assignment of a measurement to a target. To prevent a combinatorial state explosion, a measurement is only associated with a target, if the measurement lies within a certain validation region surrounding the prediction. Other optimizations are discussed in the next section. Since the mean and the covariance of the target estimation are \bar{x} and \bar{P} , we can write Eq. (2) as $v = z_k - H\bar{x}_k$, and calculate the covariance of v as:

$$B = H\bar{P}_kH^T + R \quad (4)$$

A measurement z_k lies with an " η -sigma" validation region if

$$(z_k - H\bar{x}_k)^T B^{-1} (z_k - H\bar{x}_k) \leq \eta^2 \quad (5)$$

After all hypotheses are generated, the probability of each hypothesis Ω_i^k is calculated as:

$$P_i^k \equiv P(\Omega_i^k | Z^k) \approx \prod_{m=1}^M f(z_m) \quad (6)$$

where M is the cardinality of the measurements set Z^k and

$$f(z_m) = N(z_k - H\bar{x}_k, B) \quad (7)$$

in which $N(x, P)$ denotes the normal distribution $\exp[-\frac{1}{2}x^T P^{-1}x] / \sqrt{(2\pi)^n [P]}$. The values of x and B are the Kalman filter's estimates before time k , calculated through equations (3) and (4).

When a hypothesis was chosen, the algorithm updates the projected estimate of the state variables in the Kalman filter as:

$$\begin{aligned} K &= \bar{P}H^T R^{-1} \\ x_k &= \bar{x}_k + K(z_k - H\bar{x}_k) \\ P_k &= \bar{P} - \bar{P}H^T (H\bar{P}H^T + R)^{-1} H\bar{P} \end{aligned} \quad (8)$$

where K is called the Kalman gain. The values of x_k and P_k will be used in the calculation in the next time step $k+1$.

B. Optimizations

The number of hypotheses can grow very quickly when MHT is used. To enhance the tracking speed, we use the so called zero-scan algorithm. After each time step, the zero-scan algorithm only follows the hypothesis with the highest probability, and discards all other alternatives. Still, all permutations of new location samples within the validation region assigned to existing paths have to be considered, and the corresponding probabilities have to be calculated. The number of hypotheses to be checked mainly depends on the density of the nodes in the simulation area. As we are

simulating an inner city traffic scenario, very high vehicle densities can occur and therefore large numbers of hypotheses have to be processed. In order to be able to calculate a result with limited resources, the hypotheses tree has to be reduced as much as possible. In traffic scenarios, most of the combinations are obviously not possible, as the vehicles are restricted by environment physical limitations. Samples may not be reachable from the end of a specific path given specific limitations on acceleration/deceleration or speed of vehicles are neglected. Before the hypotheses for a sample are generated, those with a probability below a certain threshold are discarded. Moreover, paths (tracks) that were not extended for some time are considered “dead” and they are no longer taken into account.

We have implemented the described algorithm and we applied it to data-sets of anonymized position samples created through various mobility models and traces. The implementation allows us to track up to 175 vehicles in real time on a common desktop computer (with a 2 GHz CPU). In our simulation environment, experiments with more than 175 vehicles result in traffic jams and thus in vehicle density that rises sharply on some road segments; this increases the delay and memory needs for tracking. We describe our evaluation setup and results next.

V. EVALUATION

In order to test the performance of our tracking approach, we have conducted extensive simulations. Our evaluation setup includes two steps: first, we use a discrete event simulator and a vehicular mobility model to generate mobility traces. Based on these traces, we create sample data containing the anonymized position samples. In the second step, those anonymized position samples are processed by our Multi-Hypothesis Tracker. The tracker’s results are finally compared to the original traces. As for the evaluation metric we use the maximum period of time the tracker was able to correctly reproduce the trace of each vehicle, averaged over all traces in the simulation.

A. Simulation and Tracking Setup

Due to the lack of real-world data covering hundreds of vehicles over a long period of time, we depend on traffic simulations to produce the tracker input. For trace generation we use the JiST/SWANS ad-hoc network simulator [1] in combination with the STRAW vehicular mobility model [5]. This combination has been found to simulate vehicular mobility reasonably well [22].

STRAW simulates vehicle movements in traffic networks composed of road segments, which are sub-divided into lanes. Depending on the type of the street, there are traffic lights and the amount of lanes in each direction and the maximum speed differs. Vehicles moving on each lane are periodically calculating the acceleration or deceleration for the next time step. The calculation considers the free space to the preceding vehicle within the actual or the next road segment. A vehicle has to wait at the end of a segment, until there is room on a

Mobility model	STRAW (urban/central Boston map)
Number of nodes	25 – 250
Fieldsize	1000 m x 1000 m
Simulation time	1000 seconds
Beacon rate	1 Hz (if not noted otherwise)
Max. node velocity	11 – 26 m/s (road dependent)
Max. acceleration	2.23 m/s ²
Max. deceleration	11.15 m/s ²
Simulation runs	10 (per number of nodes)

TABLE I
OVERVIEW ON SIMULATION PARAMETERS.

lane in the following segment. To smoothen out the vehicle movement, i.e., to avoid jumps between road segments, an intersection area is defined in-between road segments. Vehicles inside the intersection area are moving along Bézier curves from one lane end to the starting point of another. There is no collision recognition implemented, so vehicles crossing an intersection at the same time may contact each other. Moving on one road segment, vehicles cannot change lanes, thus overtaking does not happen. But vehicles can change lanes when entering a new road segment. On average, this happens every few seconds, thus they are able to pass slower vehicles on a neighboring lane.

Among the advantages of this setup are (i) the broad availability of map data (with the help of a converter, maps of the TIGER² format are supported), and (ii) the realistic simulation of vehicle behavior according to the physical movements of massive objects with limited acceleration. The fact that vehicles sometimes drive through each other on intersections makes it hard for the tracker to find the correct path in the specific situation, so, keeping a safety distance between vehicles would further improve tracking results. Table I lists the most important configuration settings for the generation of mobility traces. The resultant position data is then fed into the Multi-Hypothesis Tracker. Our MHT implementation is widely configurable and contains almost 50 different preference settings. We tracked several thousands of single simulations to learn about the influence of the different settings on tracking success. Here we only want to introduce the ones that we found to be most relevant.

The disturbance and measurement covariances (i.e. Q and R) of the Kalman filter turned out to be a determining factor for tracking accuracy. They are determined each by only one variable (q and r), and we have tested different ranges in our simulations. As shown in Fig. 2, it is mostly the ratio of these two values that determines the tracking accuracy. The tracking success is measured as the average of the duration of each path that is correctly tracked, and it can be a fraction of the total simulation duration. We found that if the disturbance covariance gets too large, one can observe strong oscillations in Kalman predictions and the results get almost random. If the measurement covariance gets too large, then there is a

²Topologically Integrated Geographic Encoding and Referencing, <http://www.census.gov/geo/www/tiger/>.

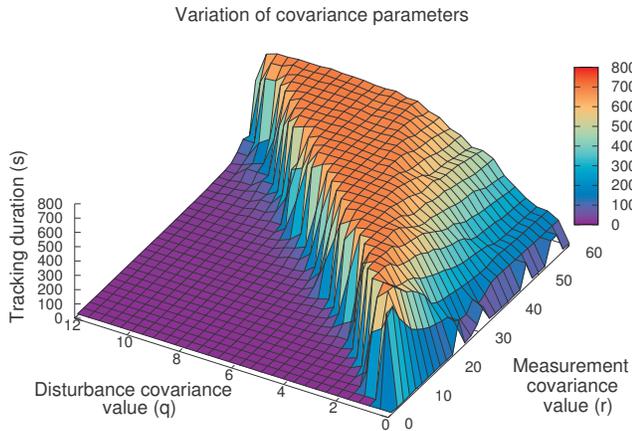


Fig. 2. Effect of Kalman measurement and disturbance covariance on tracking results.

strong attenuation of predictions and the system gets very inert. As one can see in the figure, there is a plateau with good tracking results from the point where parameters with the right ratio have to be chosen, in order to get good tracking results. Parameters need only to be adapted if, for example, the position measurements are very noisy or the behavior of the nodes becomes very unpredictable. Thus, q and r were chosen statically for most of the simulations.

Other important parameters govern the behavior of the MHT itself. In order to have an acceptable runtime performance, one has to decide where and how to prune the hypothesis tree. Too aggressive pruning reduces the tracking success, whereas too lax pruning can increase the runtime by some magnitudes, rendering tracking impractical. Another important parameter called *false_samples* is introduced to allow for false correlations of single position samples to tracks. In our simulations *false_samples* is always set to 1. This means that even though a track contains a single wrong position sample, it is still considered as correctly tracked if it then continues with the correct samples. This is needed for situations where position samples of intersecting paths are located closely together. Then, the tracker is not able to correctly distinguish which sample belongs to which vehicle. However, as the two vehicles continue their trip, the tracker is again able to tell them apart.

B. Tracking Results

In our first evaluation, we want to check what tracking results can be achieved for different beaconing rates. We vary the beaconing rate and use a fresh pseudonym for every message sent, so that the beacon messages are essentially anonymous. Fig. 3 shows that for high beaconing rates (one per second and faster) and for a lower number of vehicles the tracking duration is about 800 seconds and above. This means that on average any vehicle in the simulation is tracked for 800 seconds out of 1000. At higher vehicle densities, the average tracking success goes down to about 700 seconds. Manual

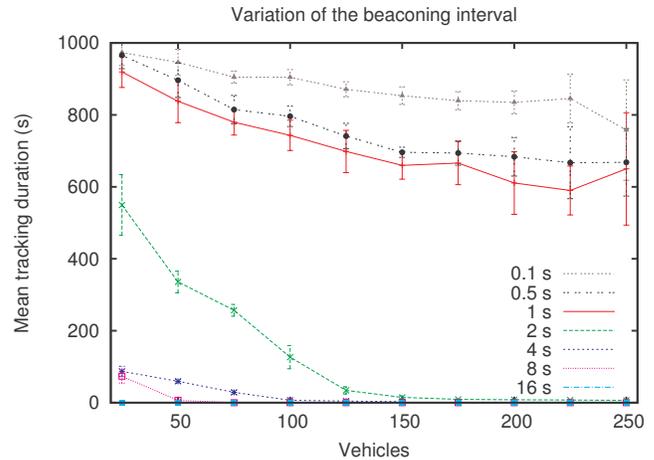


Fig. 3. Effect of IVC beaconing intervals (ranging from 0.1 to 10 seconds) on tracking results.

analysis of the simulations shows that wrong correlations occur almost exclusively at intersections, where traces are crossing each other. One could also observe that the variation between simulation runs with same parameters increases, as vehicle density increases. As higher vehicle density leads to a higher probability of multiple vehicles being at an intersection at the same time, this also increases the risk for tracking errors and is the cause for the poorer performance.

For lower beacon rates, of one beacon per 2 seconds and less, we see that the tracker performs much worse: there is simply not enough data available to reliably track vehicles through intersections. Beacon rates could be seen as one way of ensuring privacy. However, as current field trials and standardization activities assume a beaconing rate of 1 to 10 Hz, one cannot count on low beaconing rates for privacy protection.

Up to now, we have assumed that vehicles change their pseudonyms for every packet. However, current research on VANET security suggests that vehicles will only have a limited set of pseudonyms and will have to reuse each of them for multiple packets (also for practical reasons due to the IVC protocol functionality and applications) [16]. In Fig. 4, vehicles are assumed to use a constant beacon rate of 1 Hz, but to keep their pseudonym identifiers for intervals of up to 10 seconds, i.e. they reuse one pseudonym for ten successive packets before changing to another one. In reality, the reuse periods of time might even be in the magnitude of minutes. As one can see, pseudonym change intervals of 4 seconds and above lead to almost 100 % tracking success: the vehicles were tracked consistently during almost the whole simulation time of 1000 seconds. Pseudonym change intervals are assumed to be constant, but for short random time offset for each vehicle to prevent 'synchronization' effects.

Another effect could be that at least in the initial deployment phase only a small percentage of vehicles will be equipped with VANET communication units. Therefore, vehicles send-

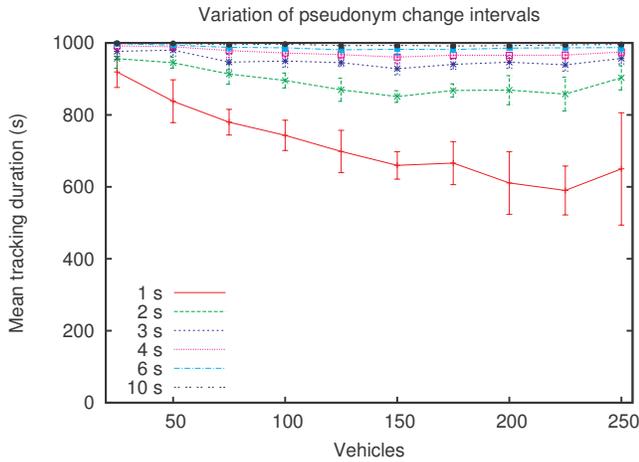


Fig. 4. Effect of pseudonym change interval (1 - 10 seconds) on tracking results.

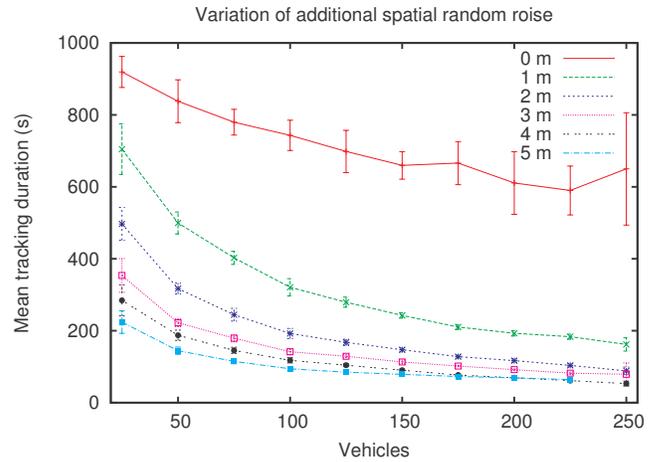


Fig. 6. Impact of noisy position information (σ in range of 0 - 5 meters) on tracking results

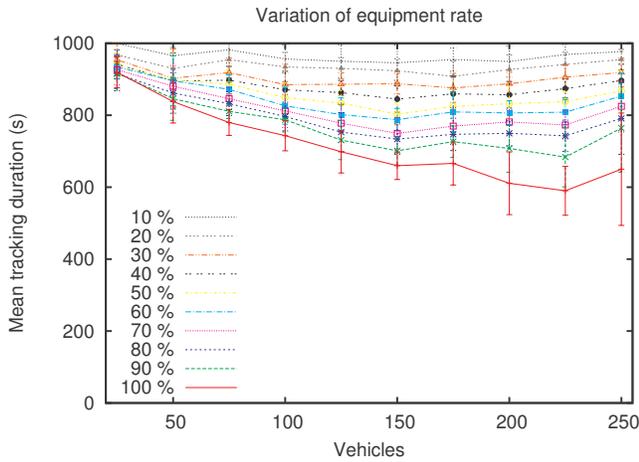


Fig. 5. Impact of equipment penetration rate (10 - 100 %) on tracking results.

ing beacon messages will be separated from other senders by a large number of unequipped cars. This facilitates the tracker (the adversary), as the tracks of equipped cars will cross only rarely. As shown in Fig. 5 low equipment penetration rates, when only 10 or 20 % of vehicles use VANET communication units, lead to an average tracking duration of more than 900 seconds. Of course, only equipped cars are tracked in such a scenario.

Finally, we want to analyze how noisy position information influences the tracking success. Assuming ordinary GPS receivers, the positions reported by cars might be off by several meters. Fig. 6 assumes a scenario where the position reported in beacon messages includes the exact coordinates plus a random, normally distributed, value with σ up to 5 meters in a randomly chosen direction. The results show that even a σ of 1 or 2 meters already reduces the tracking success of our attacker very effectively. One has to note that for many envisioned transportation safety applications, such as intersection collision detection or lane change warning, a

very high positioning accuracy is required. Therefore vehicles will very likely be equipped with high-end GPS receivers, or use other mechanisms known from navigation systems to enhance location accuracy. To use random noise for modeling the GPS error is just an approximation. Usually the deviation between the GPS and the real position does not vary randomly from beacon to beacon, but is more constant over time. Just for few situations, for example, due to interference with tall buildings in city centers, the GPS error may change more spontaneously. In case of the MHT, only varying deviation effects the tracking results, as the tracker is working with distances between locations and not on absolute positions. So with real data, in most cases the tracker results will not be severely decreased by location inaccuracy from GPS.

VI. CONCLUSION

To sum up our findings, in a scenario with vehicles sending beacon messages at 1 Hz and changing their pseudonyms every 10 seconds and having an equipment rate of 20 %, an attacker with the capabilities described in Sec. III can effectively track vehicles and their drivers with an accuracy of almost 100 % using the approach described in this paper. Lower beacon rates and spatial noise of a certain level prevent a tracker from connecting anonymous position samples to a continuous path. But this would render the majority of transportation safety applications – based on vehicular communication – useless, because they require exact position information.

In contrast to results on tracking pedestrians, discussed in Sec. II, vehicle tracking is significantly more effective. Vehicles usually move very orderly along streets and mix only at intersections and this makes the task of the attacker comparatively easier. For our admittedly strong attacker, we have shown that our tracking approach can be astonishingly effective. Of course, pseudonymous approaches for VANETs were proposed to thwart a partially present adversary. But our results raise a valid question on the effectiveness of

pseudonymous (or even anonymous) schemes in VANETs and the level of achievable location privacy protection.

For future work, we will investigate the effectiveness of other tracking mechanisms, as well as different attacker models. In particular, we consider weaker adversaries and different variants; e.g., attackers that can collect a fraction of IVC packets, or/and that they are restricted to certain areas, or/and have other limitations on the quality of information they have (e.g., time). Inversely, we will investigate how enhance data processing methods could possibly lead to effective tracking even for such lower quality location data. Another item of future investigations that our results motivate: enhanced privacy-enhancing mechanisms, as well as evaluation of various privacy-related metrics. Overall, we hope these results will contribute and inspire work that leads to a more precise understanding of and mechanisms for improved location privacy in VANETs.

REFERENCES

- [1] R. Barr, Z. Haas, and R. van Renesse. JiST: An efficient approach to simulation using virtual machines. *Software Practice & Experience*, 35(6):539–576, 2005.
- [2] A. Beresford and F. Stajano. Location Privacy in Pervasive Computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.
- [3] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House Publishers, 1999.
- [4] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy. Efficient and robust pseudonymous authentication in VANET. In *VANET '07*, pages 19–28, New York, NY, USA, September 2007. ACM.
- [5] D. Choffnes and F. Bustamante. An Integrated Mobility and Traffic Model for Vehicular Wireless Networks. In *Proc. of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, Sept. 2005.
- [6] E. Fonseca, A. Festag, R. Baldessari, and R. Aguiar. Support of Anonymity in VANETs - Putting Pseudonymity into Practice. In *IEEE Wireless Communications and Networking Conference (WCNC)*, Hong Kong, March 2007.
- [7] M. Gerlach and F. Güttler. Privacy in VANETs using Changing Pseudonyms - Ideal and Real. In *VTC2007-Spring*, pages 2521–2525, 2007.
- [8] P. Golle and K. Partridge. On the anonymity of home/work location pairs. In H. Tokuda, M. Beigl, A. Friday, A. J. B. Brush, and Y. Tobe, editors, *Pervasive Computing, 7th International Conference*, volume 5538 of *Lecture Notes in Computer Science*, pages 390–397. Springer, 2009.
- [9] M. Gruteser and B. Hoh. On the Anonymity of Periodic Location Samples. In *SPC*, pages 179–192, 2005.
- [10] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran. AMOEBA: Robust Location Privacy Scheme for VANET. *IEEE Journal on Selected Areas in Communications*, 25(8):1569 – 1589, Oct. 2007.
- [11] R. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82(Series D):35–45, 1960.
- [12] L. Buttyan, T. Holczer, and I. Vajda. On the effectiveness of changing pseudonyms to provide location privacy in VANETs. European Workshop on Security and Privacy in Ad Hoc and Sensor Networks (ESAS 2007), July 2007.
- [13] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, and J.-P. Hubaux. Eviction of Misbehaving and Faulty Nodes in Vehicular Networks. *IEEE Journal on Selected Areas in Communications, Special Issue on Vehicular Networks*, 25(8):1557 – 1568, 2007.
- [14] Z. Ma, F. Kargl, and M. Weber. A location privacy metric for v2x communication systems. In *IEEE Sarnoff Symposium 2009 (SARNOFF 2009)*, Princeton, NJ, USA, March 2009.
- [15] Z. Ma, F. Kargl, and M. Weber. Measuring location privacy in v2x communication systems with accumulated information. In *The Sixth IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS09)*, Macau SAR, China, October 2009.
- [16] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux. Secure vehicular communications: Design and architecture. *IEEE Communications Magazine*, 46(11):2–8, November 2008.
- [17] P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, and M. Raya. Architecture for Secure and Private Vehicular Communications. In *the 7th International Conference on ITS Telecommunications*, June 2007.
- [18] P. Papadimitratos, A. Kung, J. Hubaux, and F. Kargl. Privacy and identity management for vehicular communication systems: a position paper. In *Workshop on Standards for Privacy in User-Centric Identity Management*, 2006.
- [19] M. Raya and J.-P. Hubaux. The Security of Vehicular Ad Hoc Networks. In *Proc. of Third ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN 2005)*, Alexandria, USA, Nov. 2005.
- [20] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24:843 – 854, December 1979.
- [21] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki. CARAVAN: Providing Location Privacy for VANET. In *Proceedings of Embedded Security in Cars (ESCAR)*, Nov. 2005.
- [22] E. Schoch, M. Feiri, F. Kargl, and M. Weber. Simulation of Ad Hoc Networks: ns-2 compared to JiST/SWANS. In *First International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SimuTools)*, Mar. 2008.
- [23] E. Schoch, F. Kargl, T. Leinmüller, S. Schlott, and P. Papadimitratos. Impact of Pseudonym Changes on Geographic Routing in VANETs. In *Third European Workshop on Security and Privacy in Ad hoc and Sensor Networks (ESAS 2006)*, November 2006.