Let Your Reputation Precede You

[Poster] Let Your Reputation Precede You

Claudio Casetti, Stefano Tramacere
Politecnico di Torino, Italy

***Abstract*—Although Vehicular Networks are still on the drawing board, the recent announcement by the NHTSA that they will begin working on regulations for V2V (Vehicle-to-Vehicle) communication has spurred concerns that more and more personal data may be unduly disseminated as we drive our cars. We examine a scenario where vehicles use anonymous certificates, provided by a central authority, and do not divulge their position (e.g., for the purpose of accident prevention), but merely exchange contextual traffic information (congestion, roadworks, accidents in the area...). The reliability of such information is corroborated by the vehicle *reputation*, assigned by a Central Controller (CC). The mechanisms that lead to the forming of vehicle reputation are outside the scope of this paper. We are instead interested in designing a system for the robust dissemination of reputation and in understanding its implications on user privacy.**

I. System model

We assume that vehicles use Inter-Vehicle Communication (IVC) to interact with other vehicles and with RoadSide Units (RSUs), when under coverage. We also assume that RSUs do not provide a continuous coverage of the entire map, so that vehicles spend some time outside direct coverage of RSUs, although they can communicate with nearby vehicles. Vehicle identities are anonymized through the use of pseudonyms [1], i.e., short-lived certificates that are provided to vehicles and periodically refreshed when they are under RSU coverage. Certificates are used to prove that vehicle X is a legitimate part of the IVC network and, as we shall detail below, to establish the reputation of the vehicle. Upon each encounter with an RSU, a vehicle can request a certificate refill and it receives C anonymous certificates, each with a fixed lifetime T_L established by the RSU. The lifetime is necessary to simplify revocation and we assume it is generally longer that the actual certificate usage time by each vehicle. We assume that the protocol exchange that leads to a certificate refill occurs on a secure channel established between a Central Controller (CC) and the vehicle, through the RSU. In order to minimize the chance of being tracked, a vehicle uses a certificate only for a time $T_C < T_L$ at most, before discarding it. To summarize, in the i-th time interval of duration T_C, a certificate $\chi_X(i)$ is associated to a pair of private/public keys $\{k_X(i), K_X(i)\}$ that vehicle X uses to authenticate the messages it sends throughout that time interval.

The reputation of vehicles is managed by the CC and it is classified out of R classes, based on evidence of previous misconduct, or lack thereof. Beside managing the reputation, the CC can link certificates to real IDs of vehicles and thus knows: (i) which certificates belong to which vehicle; (ii) what is the current reputation class of the vehicle. Consequently, the CC can establish the reputation of any vehicles that issues a specific certificate. The goal of the system is *to enable vehicles to identify the reputation class of another vehicle they are receiving information from, without the need to divulge the identity of either vehicle*. Such a goal can be achieved in different ways, not all viable in the sparsely connected scenario we are considering, or computationally efficient. For example, the updated reputation score could be provided and signed by the CC upon every certificate refill, but then it would be up to the vehicle to use it (or misuse it). Or, the score could be cryptographically embedded (though publicly verifiable) in the certificates issued by the CC, but this solution would still leave a T_L window of vulnerability for misuse in case the reputation changes in the meantime.

The system we envision leverages the use of Bloom filters for reputation advertisement. Bloom filters are probabilistic data structures that allow a verifier to establish with certainty whether an element is part of a set, although they may yield false positives for elements not part of that set. The use of Bloom filters for VANET certificate management (specifically for their revocation) has already been advocated in the literature, e.g., by [2].

We assume that the CC periodically uses a Bloom filter to summarize the certificates currently assigned to each vehicle in a class and whose lifetime has not expired. In other words, for each of the R classes, the CC feeds each certificate of that class to k hash functions and computes a Bloom filter of size m bits. Such filters are periodically broadcasted by each RSU and received by vehicles under their coverage. Vehicles store the filters for offline use when outside RSU coverage. When vehicle A sends traffic information during a time interval i, it signs the message using the private key $k_A(i)$ associated to the (anonymous) certificate χ_i it is using in that time interval. The signed message is broadcast together with certificate χ_i. Vehicle B verifies the validity of the message (i.e., that it was broadcasted by a legitimate member of the IVC) using public key $k_A(i)$ associated to the certificate attached to the message. Subsequently, vehicle B can easily identify the reputation of vehicle A (and thus decide whether or not to trust the traffic information) by matching the certificate with one of the R Bloom filters it has stored since its last encounter with an RSU.

II. Dimensioning the system

In order to quantify the impact of our solution and its effectiveness, one of the main metrics is the *probability of multiple class matching*, p_M. Multiple class matching is a consequence of false positives in Bloom filters and it is computed as the probability that a certificate issued by vehicles A is detected by vehicle B as belonging to two or more classes (only one of them being correct).

Multiple matchings invalidate the procedure to establish the reputation. From [3], we derive the probability of false positives when n elements are inserted in *one* bloom filter of size m bits using k independent hash functions:

$$p_F = \left(1 - \left(1 - \frac{1}{m}\right)^{kn}\right)^k,$$

therefore, under the simplifying assumption that all classes have the same p_F, the probability of multiple class matching can be written as:

$$p_M = 1 - (1 - p_F)^{R-1}.$$

The above equations allow the dimensioning of key system variables, such as the size of Bloom filters and the number of certificates that can be refilled. Let us look at a practical example with some simplifying assumptions.

Road map and vehicles. Consider a 1km x 1km grid road map, consisting of 11 equispaced horizontal roads intersecting 6 equispaced vertical roads, for a total of 17 km of roads on the map. All roads are single-lane and bidirectional. Some RSUs (the exact number is not relevant in our example) provide coverage on the map and feed the Bloom filters to passing vehicles. In a mildly congested scenario, assuming that vehicles travel 10 m apart on average, we would find 1,700 vehicles on the map at any given time.

Certificates and filter refresh rate. In order to avoid tracking, vehicles use a certificate only for limited time T_C. They will therefore pre-load up to C certificates from a nearby RSU when their certificate buffer is almost empty. For example, if we assume that $C = 60$ and that each certificate is only used for $T_C = 30$ seconds before being discarded, vehicles can expect 30-minutes' worth of certificates upon each refill. The lifetime of each issued certificate can be set by the CC as $T_L = CT_C$. Thus, the Bloom filters must be suitably dimensioned so as to provide a low p_M when subject to a large intake of certificates. In particular, Bloom filters have to be frequently reset and recomputed to be able to track potential reputation changes. We will assume a one-minute filter refresh rate for every reputation class.

Number of certificates and multiple matching probability. During the one-minute refresh period, the Bloom filter of each reputation class should be loaded with all current certificates (i.e., those downloaded in the past 30 minutes) of vehicles in that class. For simplicity, we only consider vehicles that are on the map when the filter is refreshed.

Given the above parameters, and setting $k = 3$ and $R = 5$, in Fig. 1 we plot p_M as a function of the number of certificates that each car is allowed to store, for different values of filter sizes (in bytes). It can be seen that the probability of multiple matching is lower than 0.01 for the value $C = 60$ used in our previous example and the filter size is just 256 bytes, which entails that filters for several classes could be broadcast by an RSU in a single message.

III. Security analysis

It is well known in the literature that vehicle data (such as ID, position, direction) issued by beaconing as mandated by IEEE 1609.2 makes vehicles susceptible of being tracked by eavesdroppers. Even if the ID of the vehicle is anonymized through the use of pseudonyms, tracking is still considered to be possible [4], and several proposals exist to avoid explicit position advertisement in beacons. However, even in the absence of explicit position as in the system we have examined, a vehicle could still be tracked by correlating other information leaked by the vehicle. The system described above could lead to potential vehicle tracking by an internal attacker that can eavesdrop by disseminating a large number of rogue micro RSUs in the area. The attacker can then correlate segments of the map where the same certificate was used. Such an approach could however be thwarted if certificate usage times T_C were drastically reduced, although this would require the downloading and buffering of a larger batch of certificates from RSUs. As shown in Fig. 1, this can be achieved at the cost of larger filter sizes, to avoid increasing the likelihood of multiple matching. Further vehicle tracking could result from the attacker correlating the number of vehicles in each reputation class, collected by its rogue RSUs at different times. Given the system we have devised, such an attack would be independent from the certificate lifetime or number, since the Bloom filter would unequivocally link each certificate to a reputation class. The latter approach could however be thwarted if the reputation class of a vehicle can be randomly obfuscated (without excessively altering the real vehicle reputation). We leave the investigation of these issues for future work.

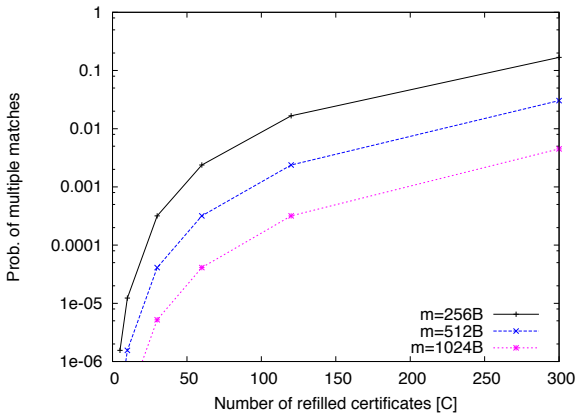

Fig. 1. Probability of multiple class matching as a function of the number of refilled certificates per vehicle, for different Bloom filter sizes.

References

[1] P. Papadimitratos et al., "Secure vehicular communications: Design and architecture," IEEE Communications Magazine, 46(11):2-8, November 2008.

[2] J.J. Haas, H. Yih-Chun Hu, K.P. Laberteaux, "Efficient Certificate Revocation List Organization and Distribution", IEEE Journal on Selected Areas In Communications, 29(3):595-604, March 2011.

[3] L. Fan, P. Cao, J. Almeida, A. Z. Broder, "Summary cache: a scalable wide-area web cache sharing protocol," IEEE/ACM Transactions on Networking, 8(3):281-293, June 2000.

[4] B. Wiedersheim, Z. Ma, F. Kargl, P. Papadimitratos, "Privacy in inter-vehicular networks: Why simple pseudonym change is not enough," IEEE WONS 2010, Kranjska Gora.