Adaptive Message Authentication for Multi-Hop Networks

Nikodin Ristanovic[1], Panos Papadimitratos[2], George Theodorakopoulos[1],
Jean-Pierre Hubaux[1] and Jean-Yves Le Boudec[1]

***Abstract*—Recent benchmarks indicate that the use of public key cryptography results in non-negligible verification times on platforms with limited processing power. In this paper, we focus on multi-hop Inter-Vehicle Communication and show that the increase in message processing time in vehicular nodes degrades network performance, by decreasing the number of messages that reach destinations. We propose Adaptive Message Authentication (AMA), a lightweight filtering scheme that reduces the number of cryptographic operations performed by the nodes. Although based on local observations and without any additional communication channel between the nodes, our scheme achieves global improvement of network performance. We perform extensive simulations and show that our scheme resists DoS attacks even against a substantial number of adversaries in the network.**

I. Introduction

It is envisioned that inter-vehicle communications (IVC) will enable Vehicular Ad hoc Networking (VANET). Short-range wireless IVC will be based on a variant of the currently widely used IEEE 802.11 protocol. The IEEE 1609.x protocol suite, also known as the WAVE technology [1] developed for the U.S. Department of Transportation (DOT), is already at the stage of a trial-use standard. Vehicles will communicate with other vehicles within range, but, equally important, they will cooperate in forwarding messages of their neighbors, other vehicles or road-side units (RSUs), across multiple hops.

Vehicular communications (VC) will be able to support various applications, among which transportation safety and transportation efficiency play a prominent role. Vehicles using congestion notification applications [2], [3], [4] will collect data, create useful information and then send it across multiple hops towards a "targeted" or "destination" geographic area. The receiving vehicles in the destination area will utilize these notifications, possibly to alert their drivers to avoid dangerous areas or optimize their routes.

Security is an important requirement for this manner of forwarding. In recent years the foundation for securing vehicular communications has been laid out in the literature [5], [6], [7] and a substantial effort in academia and industry has been invested in providing adequate solutions; for example, the IEEE 1609.2 trial-use standard [8], the Network On Wheels (NoW) project [9], and the Secure Vehicular Communications (SeVeCom) project [10]. These prominent efforts are based on well-established cryptographic building blocks, such as public key cryptography, and Certification Authorities (CAs), and they mandate that all VC messages be digitally signed and carry the certificate of the signing node (vehicle or RSU). This way, without any association with the signer, the authenticity of any message can be validated by any receiving node.

Although very convenient for use in vehicular networks, public key cryptography is costly and introduces significant processing overhead. Recent benchmarks, such as those obtained within the framework of the European eCrypt project [11], show that signature verification on a wide range of computing platforms takes a significant amount of time, even for the fast elliptic curve algorithms proposed for use in vehicular networks [8], [9], [10]. Due to the on-board vehicle equipment cost constraints, the currently envisioned automotive communication boxes face the same limitations: cryptographic message processing delays are typically in the order of several milliseconds [12]. More importantly, with tens of nodes (vehicles) usually in proximity, each node has to handle and validate hundreds of messages per second.

We argue that the processing overhead in intermediate nodes can result in decreased network performance, due to the limited processing capabilities of the envisioned vehicular platforms. Our goal is to decrease the number of cryptographic operations performed by the nodes and to avoid deterioration in performance due to processing power limitations. At the same time, we verify that this reduction of message verifications does not make a vehicular network more vulnerable to outside adversaries, or to DoS attacks.

In this paper we focus on traditional approaches to identity management and secure inter-vehicle communication in vehicular networks, such as [13]. In terms of Inter-Vehicle Communication and multi-hop forwarding, these proposals recommend two extreme strategies. The first group of proposals requires intermediate nodes to verify that messages originate at legitimate senders and to check the integrity of the messages before resending them. We show that this approach to securing multi-hop forwarding tends to be too pessimistic and results in many unnecessary message verifications, degrading the network performance. The second approach, on the contrary, advocates skipping message verification in intermediate nodes and neglects nodes' vulnerability to DoS attacks. Although it performs well with few adversaries in the network, our simulations show that when no message verifications are performed, the goodput of legitimate nodes significantly drops as the number of adversarial nodes in the network increases.

The solution we propose is an adaptive scheme that integrates the best features of the two aforementioned approaches. The aim is to make nodes perform only the necessary number of cryptographic operations while skipping the redundant

978-1-61284-188-5/11/$26.00 ©2011 IEEE

message verifications and improving the overall performance of the network. The scheme takes advantage of the fact that nodes in different parts of a vehicular network face different security conditions at a given point in time. For instance, nodes in less hostile areas can afford to be less cautious (check fewer messages) than others. However, given the dynamic nature of vehicular networks, the situation may change quickly and dramatically, so nodes should have the ability to adapt to changing circumstances.

Our contribution is twofold:

- We propose AMA (Adaptive Message Authentication), a scheme that probabilistically checks messages in intermediate nodes. Our scheme is reactive in that the checking rate increases to 100% only when forged messages are detected, and only for a limited period before returning to probabilistic checks. AMA is independent of the forwarding algorithm or the wireless standard used for communication and it can be easily integrated in the existing frameworks for secure communications in vehicular networks.
- We show through extensive simulations that the scheme guarantees substantial performance gains over the traditional proactive approach. The adaptiveness of the scheme increases the performance in cases with few adversaries, as well as in the cases when the adversarial nodes represent a significant percentage of the population.

The paper is organized as follows. In §II we present the example vehicular applications of interest, our motivation and the intuition behind our approach. In §III we describe the system and the adversary model that we are dealing with in this work. In §IV, we define rigorously the core of the problem we are solving. We provide the complete description of AMA in §V, and its evaluation in §VI. We discuss possible extensions of AMA in §VII. We present related work in §VIII, and we conclude in §IX.

II. Motivation and Intuition

In this section, we pinpoint why a group of VANET protocols and applications call for a new practical solution, such as the AMA we contribute here.

Multi-hop vehicular communication is considered in several transportation safety and efficiency applications, in particular *congestion notification* and *environmental hazard notification* applications [2], [3], [4]. These applications exploit the inter-vehicle communication and more specifically *GeoCast* or *Position-based* routing. Their messages contain the destination or target geographic area, within which all receiving vehicles should benefit from the information inside the messages.

Adversaries can try to degrade the performance of such a system by creating forged messages alleging they bear useful information (notifications, warnings, etc.). The danger is twofold: (i) the use of forged messages at destinations and (ii) reduction in goodput (expressed in the number of legitimate messages that reach their destinations) caused by flooding of the network with forged messages. The first risk is not really a concern, as the scheme we propose requires nodes to check each message in the destination region, prior to its use. The impact of the second threat depends critically on the ability of the security solution to prevent the spatial propagation of forged messages. We address this issue by applying an adaptive scheme that becomes very conservative as soon as a threat is detected. If an adversary tries to inject messages with its current location as the destination region (destination zone), it will be detected by all the first-hop neighbors, because the scheme we propose checks all the messages received in the destination region. In case an adversary selects a region (other than its current location) as the destination, the scheme fights this danger not only locally around the adversary, but all the way to the destination region, and if needed, inside the destination region itself.

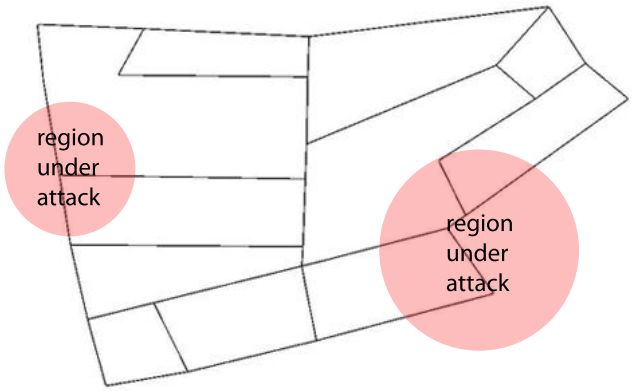

Fig. 1. A few regions in the city are under attack (shaded areas) and nodes in these regions actively defend. Nodes in the rest of the city, where the presence of the adversaries is not felt, can relax their security. We use this road map in all of our simulations.

When a threat is not present, our scheme relaxes security and avoids putting unnecessary processing load on resource-limited on-board equipment. We show that this does not lead to a reduction in goodput due to a possible rise in number of forged messages in the network. The goodput is actually increased, as processing bottlenecks caused by an overpessimistic approach in fighting the security threats are avoided.

The basic idea we draw on is based on the observation that an adversary can in practice have only a limited physical presence [14], [7]; adversaries can select any part of a vehicular network and place it under attack, but they can only be in a limited number of locations at any given time. Fig. 1 illustrates this; it shows a part of a city with the regions in which the adversarial nodes are present (shaded areas).

So, if the security conditions in different parts of the network are completely different, why would the nodes in these areas behave in the same manner? In other words, a node should respond to a threat only when it is affected by an attack and should reduce its defensive activities when the threat is not present, assuming that these activities consume resources and cause the node to underperform. With this approach, as we show with our solution, the performance of a network as a whole increases, as nodes perform only the necessary

cryptographic operations but skip the redundant ones.

III. System and Adversary Model

Vehicular Communications (VCs): We consider a system with vehicles equipped with a variant of IEEE 802.11, on-board sensors, and a computing platform. The vehicle on-board equipment, which we term a node, allows for the exchange of information among vehicles and roadside infrastructure and enables a range of applications, notably including *safety* (e.g., accident alerts) and *traffic efficiency* (e.g., distributed congestion notification) [15], [16]. VC-enabled applications entail two basic types of communication: (i) one-hop high-rate safety messaging (beaconing), with messages bearing information on the location, speed, acceleration and heading of the sender, transmitted at a rate of 3 to 10 beacons per second, and (ii) multi-hop communication between any sender disseminating information to one or more receivers, identified primarily by their location.

Safety beaconing rates are specified by standards, and the processing of those beacons is obligatory. A node must maintain a fine-grained knowledge of the motion dynamics of other vehicles in its vicinity in order, for example, to be able to detect a danger from a slowly moving vehicle ahead or one that attempts a left turn. Multi-hop communication serves, in general, less time-critical applications, and the amount of traffic depends on the supported application(s), and in principle any node could initiate such a transmission.

We define L to be the set of legitimate users running applications enabled by multi-hop communication and N_i as the number of legitimate messages received by destination i over the time period of interest. We consider here multi-hop transmissions originating at each node at a constant rate of r_L messages per second. Then, in the presence of any communication impairments and networking faults and delays, we define the *goodput* γ_L as:

$$\gamma_L = \frac{1}{|L|} \sum_{i \in L} \frac{N_i}{\text{total time}} \tag{1}$$

Position-based communication protocols appear as a natural choice for multi-hop communication, as nodes are expected to be aware of their own location (through GPS, or other localization techniques with respect to terrestrial infrastructure), and the highly volatile topology would make other mobile ad hoc networking protocols inefficient in this setting. Such algorithms, largely termed geocast protocols, are currently under consideration towards standardization [4]. As the exact implementation of the communication is largely orthogonal to our paper, we consider here two representative position-based algorithms. The first algorithm, Cached Greedy GeoCast (CGGC) [17] belongs to the group of beacon-based unicast routing algorithms. CGGC relies on beacons to discover the position of neighboring nodes (within the nominal communication range), and then forwards messages in the geographic direction of the destination, picking the node whose coordinates are the closest to the destination. If the local optimum is reached, the message is added to the local cache, where it is kept until a suitable next hop is found. The second algorithm we consider is Contention-Based Forwarding (CBF) with the basic suppression scheme based on timers [18]. Unlike CGGC, CBF is based on broadcast and performs greedy forwarding without the help of beacons and neighbors' tables. We believe that this selection of routing algorithms covers two major groups of geocast routing algorithms proposed for vehicular networks. The first group is composed of routing algorithms that heavily rely on beacons and neighbors' tables in the process of suitable next hop selection. The second group contains CBF algorithms with different suppression schemes that leave the next hop selection process and the forwarding decision to the neighbors in the transmission range.

Security Assumptions: As proposed in [9], [8], [10] the node identities are managed by a Certification Authority (CA). Each node in the network is assigned a set of private/public key pairs, with the latter being certified by a CA. Each legitimate node registered with the CA can participate in the VC system operation. Basically, nodes use their private keys to digitally sign messages they generate and the public keys of the senders of messages they receive to validate the senders' signatures.

In accordance with the state of the art in secure VC [8], [10], we assume that all messages are signed with an Elliptic Curve Digital Signature Algorithm (EC-DSA), and that for each message the certificate of the sender is attached to enable validation of the message. The validation includes first the validation of the certificate (by validating the CA's signature), and then the validation of the sender's signature.

In the rest of the paper, we refer to a message validation as "message checking." We denote the *processing delay* needed for a message to be checked by t_C. The value of this delay depends on the processing power of the on-board platform. For platforms similar to those currently considered for the proof-of-concept implementations of VC systems, characteristic delays are provided in [12], [19], whereas for other efforts PowerPC based platforms are used (e.g., DENSO platform [20]), with representative processing delays available in the eCrypt project benchmarks [11].[1] Lastly, the IEEE 1609.2 efforts currently consider, for proof-of-concept purposes, hardware-accelerated signature verification at several milliseconds. In all existing and upcoming systems, the cryptographic processing delays constitute a bottleneck. For example, even for the most powerful of those platforms [12], t_C would be 7.2ms; if we consider this value in a rather favorable environment, with 20 neighbors beaconing at the lowest possible rate of 3 beacons/second, by multiplying the three numbers, we obtain as a result that 43.2% of the CPU time would be devoted to message checking. In denser network settings, e.g, four-lane highways, and with the wireless medium impairments taken into consideration, as well as specific optimizations, the average processing load is still at the limit of the processor for safety traffic only.

[1] On 533MHz CPU Power PC platform, signature verification for EC-DSA with 192-bit curve (nist-p-192), requires 9 ms on average.

Adversary Model: In this paper our focus is on the *external* adversaries, that is, adversarial nodes that do not have in their possession system credentials (certificates issued by the CA). Each such adversarial node is a computing platform that can fabricate and inject messages, but cannot sign on behalf of a legitimate node. The direct goal of adversaries is to reduce the goodput γ_L of legitimate nodes by injecting the forged messages and making legitimate nodes waste their processing time on forged message verification. All adversarial nodes inject forged messages at a rate r_A messages per second.

We assume that the adversaries are aware of their number in relation to the number of legitimate nodes in the network and we define a as the percentage of adversaries in the network. Knowing a, the adversaries choose their sending rate r_A in order to minimize γ_L.

We emphasize that none of the forged messages injected by an adversarial node can be perceived as valid if it is checked by a legitimate node. Nonetheless, the stress imposed by the need to validate those messages is exactly what can lead to a DoS attack. For such an attack, any PDA or laptop can be used, and no tampering with hardware that stores the vehicle's cryptographic keys [10], [7] and no cryptanalytic attack are necessary. By preventing even a small fraction of legitimate traffic from reaching its destination, the adversary can prevent reception of messages in an area within the necessary deadlines: For example, consider road condition information that cannot be validated in the targeted geographical area, resulting in traffic jams; or, consider increased loads from fabricated traffic that prevent a node from performing safety related operations.

IV. Problem Definition

The usage of geocast routing algorithms implies that messages can traverse several intermediate hops before reaching the destination region. A logical question that arises is what strategy an intermediate node should adopt with regard to checking a large group of messages that are only relayed by that node. It is not clear whether a message that requires relaying should be checked by an intermediate node or just resent without any prior verification.

We define "check-all" and "check-nothing" as two extreme approaches that can be applied to relayed messages. "Check-all," as mentioned in Section I, is the default strategy in the existing proposals and it assumes checking of each relayed message, whereas "check-nothing" assumes that none of the relayed messages are checked. Both approaches perform well under certain circumstances, but underperform in other cases. Our goal is to provide a scheme that provides good performance for a wide range of values of a (percentage of adversaries in the network) and r_A (their sending rate) and to test it with the state of the art in vehicular routing algorithms.

The "check-all" strategy guarantees the fewest forged messages in the network, as it contains them locally and prevents their propagation. Given unlimited processing power in each node (implying a negligible checking time t_C), checking each relayed message would be the best strategy. In this case, less time would be spent forwarding the forged messages and the limited wireless capacity would be used only for forwarding the valid messages.

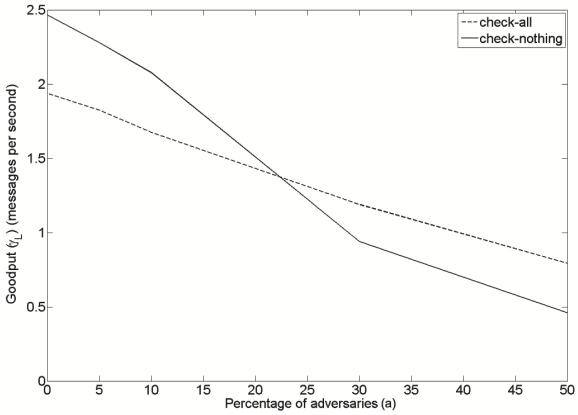

Fig. 2. The performance of "check-all" vs. "check-nothing" algorithm obtained for CBF geocast routing algorithm.

The "check-nothing" strategy promises good results with few adversaries in the network, or with few injected forged messages in the network (Figure 2). In this case, not checking any relayed traffic guarantees that no time is wasted on signature verifications in intermediate nodes and the goodput γ_L of legitimate users is improved. However, an increase in the number of adversaries in the network quickly makes this strategy inferior to "check-all".

In conclusion, "check-all" is not the best approach for networks with few adversaries and "check-nothing" gets worse as the number of adversaries increases. We want a scheme that performs well in both cases. It should contain the forged messages locally (as "check-all") in the presence of adversaries and behave as "check-nothing" with no adversaries around.

V. AMA - Adaptive Message Authentication

The reasoning behind our scheme is based on the observation that the adversaries are limited in scope and that they cannot keep the whole network under attack at all times. Consequently, we designed a scheme that is shown in Figure 3. We call it AMA (Adaptive Message Authentication).

AMA has two modes of operation. We call them "check-all" and "relaxed". The "relaxed" mode allows nodes to pay less attention to defensive measures. All the legitimate nodes are initially in the "relaxed" mode. It is this mode that is expected to improve the performance of the scheme, as only a fraction of received messages are checked by a node in the "relaxed mode". Nodes distinguish between the messages that have the current location of the node as the destination zone and those that only have to be relayed to others. Each message in the first group is checked with probability 1 and each message in the second group with probability p. If they happen to check a forged message, the forgery is always detected and it forces the node to switch its mode of operation to "check-all".

The "check-all" mode is conservative and it mandates checking each received message. A legitimate node is expected to be in this mode when there are adversarial nodes nearby. A node stays in the "check-all" mode until it receives c consecutive legitimate messages. Then, it switches back to the "relaxed" mode.

The rationale is that if a node senses that the current "temperature" of the neighborhood is low (no adversaries in the neighborhood), a node can relay most of the messages without prior authentication and integrity check, while checking only a small fraction of these messages in order to ensure a timely detection of security threats. While the selected messages are being checked, the other messages that need to be relayed do not have to wait before being forwarded.

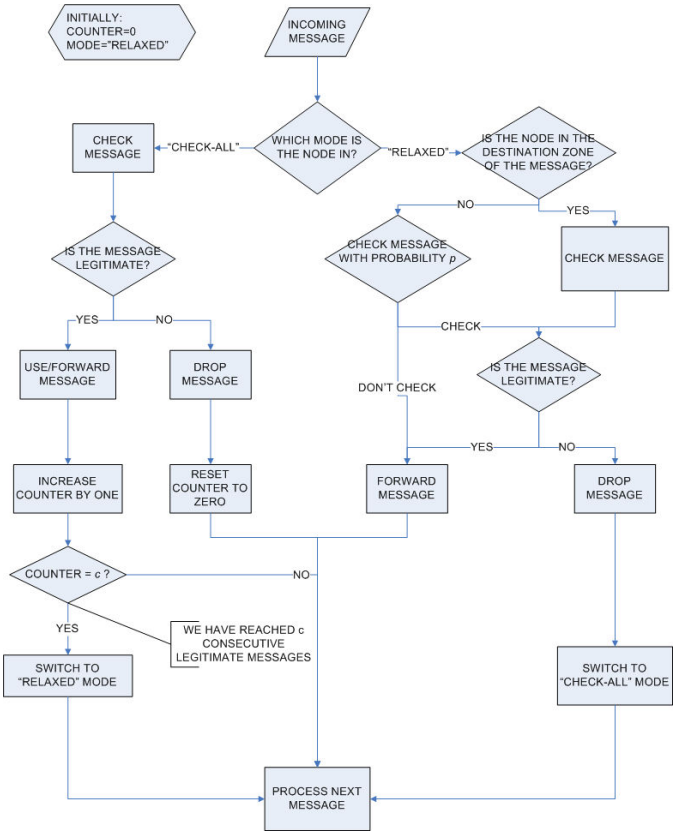

Fig. 3. AMA - the scheme for adaptive authentication and integrity checking of messages exchanged between vehicles. Briefly, an AMA node can be in one of two modes: "check-all" and "relaxed." A node starts in "relaxed" mode. In this mode, a node checks with probability 1 the messages destined for itself, but only with probability p the messages destined for other nodes. If it detects a forgery, the node switches to the "check-all" mode. In the "check-all" mode, a node checks all messages with probability 1, and switches to "relaxed" mode only if c consecutive legitimate messages are received.

It is possible, of course, to use a different function for the checking rate increase, not just a step function. We show that even this simple scheme guarantees significant performance gains, for an appropriate choice of the parameters p and c, under very realistic assumptions (the scheme and both parameters p and c are known to the adversary).

VI. Evaluation

A. Simulation Setup

The goal of the experimental validation is to compare the goodput γ_L (defined in Section III) achieved when AMA, "check-all," and "check-nothing" are used. The three strategies are tested on top of two state-of-the-art routing algorithms for vehicular networks. These are Cached Greedy GeoCast (CGGC) [17] and Contention Based Forwarding (CBF) with the basic suppression scheme based on timers [18].

For generating mobility traces, based on network topologies obtained from real maps, we use the SUMO traffic simulator (v 0.9.8) [21] with the TraNS extension [22], [23]. Vehicular mobility traces, generated for a specific road topology, are then passed to the SWANS network simulator [24].

The area that we use for our simulations is about 6 sq. km in size and it is shown in Fig. 1. It is populated with 600 vehicles that follow routes obtained from the SUMO traffic simulator with speeds that are within the legal speed limits for the observed road group. The full TCP/IP stack is implemented in each node with the exception of the transport layer. At the physical layer we use the two-ray pathloss model, which incorporates ground reflection. At the link layer, 802.11b is used and the range is limited to 200m. The SWANS implementation of 802.11b includes the complete DCF function, with retransmission, NAV and backoff functionality.

For the "check-all" and "check-nothing" approaches we run 20 simulations for every combination of percentage of adversaries $a \in \{0, 5, 10, 30, 50\}$ and adversaries' sending rates $r_A \in \{0, 1, 2, 5, 10\}$ messages per second. For AMA we run 20 simulations for every combination of percentage of adversaries $a\%$, adversaries' sending rates r_A, and AMA parameters $p \in \{0.05, 0.1, 0.2, 0.3, 0.5\}$ and $c \in \{20, 40, 60, 80, 100\}$. For each run, we randomly select $a\%$ adversaries out of the total set of nodes in the network. For each generated message, a destination region is selected at random. The legitimate node sending rate r_L is 1 message per second. The size of each geocast message is 300 bytes. The checking time t_c is $10ms$. Each simulation lasts for 500 seconds of simulation time.

Beacon verification is extremely important for security; as explained in Section III we assume that nodes check all received beacons in the simulated beaconing based routing algorithm (CGGC). As beacons make a large percentage of the total traffic in all the beaconing-based routing algorithms, we want to be as realistic as possible when simulating the load that beaconing traffic puts on the processor. The CGGC beaconing rate that we use in our simulations is 1 beacon/300ms, i.e., the value likely to become part of the standard.

B. AMA Parameter Selection

Having calculated, through the simulations, the goodput γ_L achieved under AMA for all combinations of the parameters (p, c, a, r_A), we now show how to select the parameters p and c to maximize it. This selection would be done by the network administrator before deploying the scheme in the network. In making this selection, we have to keep in mind two things:

- The percentage a of adversaries is fixed, but may or may not be known to the administrator. Below, we distinguish two cases according to whether it is known or not.
- The adversaries will learn the selected values of p and c, and choose their sending rate r_A to minimize the goodput.

In the first case, which we call the *pessimistic* case, we do not know a. So, we will select the p and c that maximize the resulting γ_L against a worst case combination of a and r_A.

$$(p^*, c^*) = \text{argmax}_{(p,c)} \min_{(a,r_A)} \gamma_L(p, c, a, r_A) \quad (2)$$

To visualize this selection, consider the following table, where each entry is equal to the goodput achieved with the corresponding row-column combination of parameters:

	(a^1, r_A^1)	$\dots$	(a^1, r_A^k)	(a^2, r_A^1)	$\dots$
(p^1, c^1)	γ_L^{1111}	$\dots$	γ_L^{111k}	γ_L^{1121}	$\dots$
(p^1, c^2)	γ_L^{1211}	$\dots$	γ_L^{121k}	γ_L^{1221}	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$
(p^1, c^m)	γ_L^{1m11}	$\dots$	γ_L^{1m1k}	γ_L^{1m21}	$\dots$
(p^2, c^1)	γ_L^{2111}	$\dots$	γ_L^{211k}	γ_L^{2121}	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

First, Eq. (2) selects the minimum element in each row. The minimum element in a row is the goodput value that will be achieved if we select the (p, c) pair of that row, and the adversaries' percentage happens to be the one in the minimizing column. If the adversaries' percentage is the right one (worst case scenario for the legitimate users), they can choose their sending rate to achieve the worst case goodput.

Then, among the minimum elements found, Eq. (2) chooses the largest one by selecting the appropriate (p, c) pair. This way, we can guarantee the adversaries would not achieve a lower goodput, even if they could change their percentage.

In the second case, which we call the *optimistic* case, we know a but not r_A. So, for the given value of a, we choose p and c that maximize γ_L against the worst case reply r_A.

$$(p^*, c^*)(a) = \text{argmax}_{(p,c)} \min_{r_A} \gamma_L(p, c, a, r_A) \quad (3)$$

Referring to the previous explanatory table, Eq. (3) now does the same minimization-maximization as Eq. (2), but operates on the columns corresponding to the known value of a. In either case, the adversaries choose their sending rate r_A to minimize γ_L, given the legitimate users' choice of p^* and c^*:

$$r_A^*(a) = \text{argmin}_{r_A} \gamma_L(p^*, c^*, a, r_A) \quad (4)$$

$$r_A^*(a) = \text{argmin}_{r_A} \gamma_L(p^*(a), c^*(a), a, r_A) \quad (5)$$

Note that the adversaries do not optimize over the percentage a, as they cannot change it.

C. Performance Evaluation

AMA outperforms "check-all" and "check-nothing" strategies for all the considered values of a, regardless of the routing algorithm (CBF in Figure 4, CGGC in Figure 5). The curves in the figures are the mean value of 20 simulations and the error bars extend a standard deviation above and below the mean values. Note that the knowledge of a (which is not easy for the administrator to obtain) guarantees only a slight performance improvement (the pessimistic scheme performs almost as good as the optimistic).

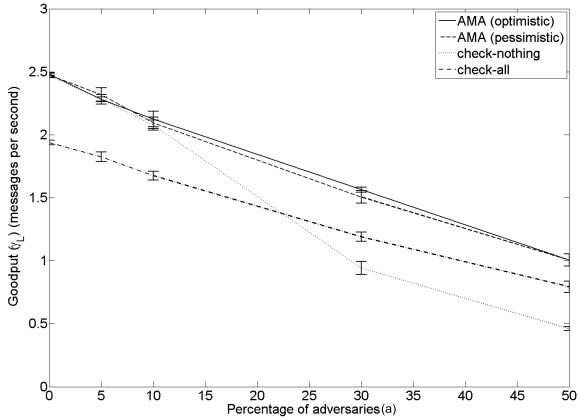

Fig. 4. Goodput γ_L obtained under (pessimistic and optimistic) AMA, "check-all", and "check-nothing" for the CBF routing algorithm.

As we can see from the figures, in the pessimistic case, which assumes no knowledge about the number of adversaries or their sending rate, the goodput of legitimate nodes γ_L improves up to 30% for CBF and up to 33% for CGGC.

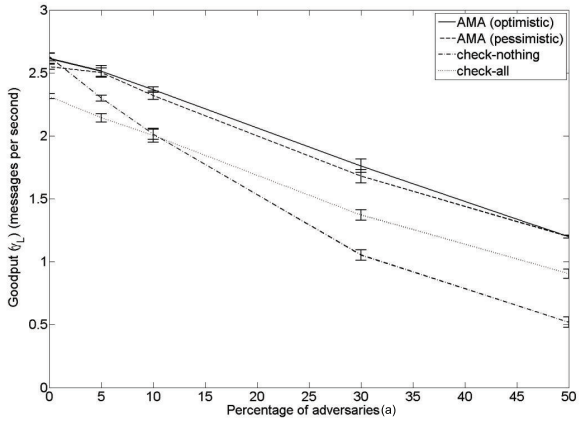

Fig. 5. Goodput γ_L obtained under (pessimistic and optimistic) AMA, "check-all", and "check-nothing" for the CGGC routing algorithm.

In the case of CBF, the performance gain is due to the reduction in the number of messages checked in the intermediate nodes. Our simulation data shows that the number of checked geocast messages drops up to 46% percent in this case.

Apart from the drop in the number of cryptographic operations, the performance of CGGC routing algorithm is affected by the beaconing processing load. Geocast messages now share the CPU time with beacons. Checking or not checking a geocast message or a group of messages can make the difference between an immediate check of an arriving geocast message and the prolonged stay of that message in

an intermediate node due to the CPU busy period introduced by beacons. The same applies to forged messages, as their increased number in the incoming queue can make valid messages wait for a period of time before being checked and relayed. This is the main reason the performance of "check-nothing" strategy drops with the increase in the number of adversaries in the network.

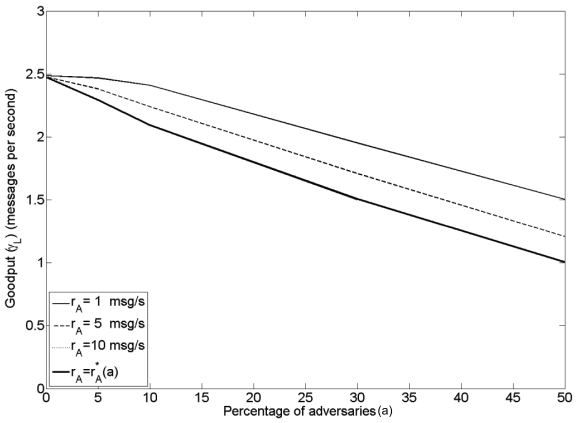

Fig. 6. Goodput γ_L obtained for various adversarial sending rates r_A under pessimistic AMA for the CBF routing algorithm. The thick line corresponds to the optimal sending rate r_A^*.

The introduction of other CPU tasks (not related to forwarding) would make this effect even more visible. If the security unit is not a stand-alone unit and has to share the CPU time with other tasks, the moment when an incoming message receives its share of CPU becomes extremely important. A single forged message or an unnecessary check of a legitimate message can make the arriving geocast message wait for an additional few hundred milliseconds due to the CPU multitasking.

Percentage of adversaries	CBF		CGGC	
	p	c	p	c
$a = 0$	0.05	60	0.2	80
$a = 5$	0.05	20	0.05	80
$a = 10$	0.2	100	0.3	80
$a = 30$	0.3	100	0.2	40
$a = 50$	0.2	40	0.2	40

TABLE I
THE OPTIMAL PARAMETERS p AND c FOR *optimistic* AMA.

In the pessimistic case, the optimal values of p and c obtained as explained in VI-B are $p = 0.2$ and $c = 40$ for both CGGC and CBF. In the optimistic case, the obtained values for p and c, for both considered algorithms, are shown in Table I.

If the adversaries have less knowledge than we assumed (i.e., if they do not know the p^* and c^*), they may choose a sending rate other than the computed optimal r_A^*. In Figures 6 (for CBF) and 7 (for CGGC) we plot the resulting γ_L-a curves for all sending rates r_A for the pessimistic choice of p^* and c^*. We see that a suboptimal selection of the sending rate by the adversaries results in improved performance for our scheme. The thick line in the figures corresponds to the optimal (for the adversaries) sending rate r_A^* and represents the worst case scenario (i.e. the lower bound for the goodput).

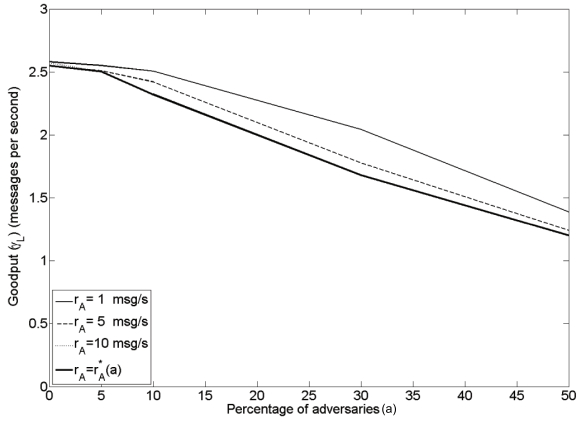

Fig. 7. Goodput γ_L obtained for various adversarial sending rates r_A under pessimistic AMA for the CGGC routing algorithm. The thick line corresponds to the optimal sending rate r_A^*.

VII. DISCUSSION

Two extensions for AMA can trade off simplicity for performance. First, the transition between two modes of AMA can be smoother than jumping from a fixed checking probability p to 1. That is, p can increase when a forged message is detected, and decrease when a legitimate message is checked. Second, depending on the application's security requirements, not all messages need to be checked by their destinations. If AMA is also applied to the messages received in the destination zone, the network performance could improve.

VIII. RELATED WORK

Three major efforts to design security and privacy enhancing solutions for VC have been undertaken in industry and academia, with the endorsement of authorities: the NoW project [9], the IEEE 1609.2 working group [8], and the SeVeCom project [10]. The basic ideas that transcend these projects are the following: (i) They build on top of the currently accepted vehicular communication protocol stack that includes safety beaconing, (ii) they rely on a *Certification Authority (CA)* and public key cryptography to protect V2V and V2I messages, with consensus on the use of Elliptic Curve DSA [25], (iii) their objective is to satisfy requirements such as message authentication, integrity, and non-repudiation (albeit in a "lighter" manner for the IEEE effort), and (iv) their intention is to protect private user information. We note that our scheme is oblivious to the exact use of certificates and public keys. As a result, AMA can remain fully operational and effective even if privacy enhancing algorithms with multiple certified public keys (pseudonyms) are implemented. Another

recent, currently on-going effort in securing VC, is the Car-2-Car Communication Consortium Security Working Group [4].

Prior to these efforts, a number of works outlined challenges [6], described attacks [7], and offered solutions [26], [12] in the field of security and privacy of vehicular networks. The last two works consider cryptographic mechanisms other than public key cryptography; in the former one, [26], symmetric key cryptography complements the public key operations to reduce overhead. The latter [12] uses group signatures to complement public key cryptography. Investigating those variants, with the somewhat different processing loads, in the context of AMA would be an interesting point for future work.

The reduction of overhead for transportation safety applications is investigated in [12], with simple, context-agnostic overhead reduction schemes, and also in [27], [28], which propose context-specific strategies for overhead reduction. The investigation of the vehicular communications security overhead and its effect on system/application performance is extended in [29], which considers both safety and efficiency applications and additional security mechanisms. These works are complementary to AMA and their joint investigation with AMA would be another interesting point for future work.

A good survey of geographic routing protocols can be found in [30]. Regarding security aspects of geographical routing, Leinmüller *et al.* [31] analyze, first, the effects of false position information on geographical routing and, then, plausibility checks for reducing the impact of wrong position advertisements. More recently, a comprehensive design and performance evaluation for secure GeoCast, including experimental implementation, has been provided in [19].

IX. Conclusion

Strict security requirements for vehicular communications led to several proposals that consider authentication and integrity checking of each relayed message as necessary conditions for secure multihop inter-vehicle communication. This default approach increases considerably the security overhead. Complex cryptographic operations, such as signature verification, introduce non-negligible processing delays. We show that the measured processing times on low-end platforms can result in the degradation of network performance. Nevertheless, ignoring security can lead to DoS attacks and even more severe decrease in network performance.

In this paper, we demonstrate that a simple, yet adaptive, filtering scheme that allows nodes to judiciously decide when to check the received message that requires further relaying, and when to simply forward it without any delay, significantly improves performance. The scheme, AMA, treats multihop messages in a reactive rather than a proactive way and requires checking of relayed messages only in the presence of a threat. Our simulations with state-of-the-art vehicular routing algorithms demonstrate that, as a result of security overhead reduction, the goodput of legitimate nodes increases up to 33%. We believe that, because of the possible significant gains, this approach is worthy of further investigation.

References

[1] IEEE1609.1, "IEEE trial-use standard for wireless access in vehicular environments (WAVE) - resource manager," October 2006.

[2] T. Shinkawa, T. Terauchi, T. Kitani, N. Shibata, K. Yasumoto, M. Ito, and T. Higashino, "Technique for information sharing using inter-vehicle communication with message ferrying," in *MDM 2006*, Nara, Japan.

[3] http://www.aqualab.cs.northwestern.edu/projects/C3.html.

[4] http://www.car-2-car.org/.

[5] M. E. Zarki, S. Mehrotra, G. Tsudik, and N.Venkatasubramanian, "Security issues in a future vehicular network," in *EW 2002*.

[6] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *HotNets-IV*, 2005.

[7] P. Papadimitratos, V. Gligor, and J.-P. Hubaux, "Securing Vehicular Communications - Assumptions, Requirements, and Principles," in *ESCAR 2006*.

[8] IEEE1609.2, "IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages," July 2006.

[9] M. Gerlach, A. Festag, T. Leinmüller, G.Goldacker, and C. Harsch, "Security architecture for vehicular communication," in *WIT 2005*.

[10] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger M. Raya, Z. Ma, F. Kargl, A. Kung, and J.-P. Hubaux, "Secure Vehicular Communication Systems: Design and Architecture," *IEEE Communications Magazine*, vol. 46, no. 11, pp. 100–109, Nov. 2008.

[11] D. Page, D. J. Bernstein, and T. Lange, "Report on ebats performance benchmarks," European Network of Excellence in Cryptology, Tech. Rep. IST-2002-507932-D.VAM.9, March 2007.

[12] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "Efficient and robust pseudonymous authentication in vanet," in *ACM VANET*, Montreal, Quebec, Canada, 2007.

[13] P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, and M. Raya, "Architecture for secure and private vehicular communications," in *IEEE ITST*, Sophia Antipolis, France, Jun. 2007.

[14] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in vanets," in *ACM VANET*, 2004.

[15] F. Dötzer, T. Kosch, and M. Strassberger, "Classification for traffic related inter-vehicle messaging," in *ITST*, 2005.

[16] N. Shibata, T. Terauchi, T. Kitani, K. Yasumoto, M. Ito, and T. Higashino, "A method for sharing traffic jam information using inter-vehicle communication," in *MobiQuitous*, 2006.

[17] C. Maihöfer, R. Eberhardt, and E. Schoch, "CGGC: Cached greedy geocast," in *WWIC*, 2004.

[18] H. Fuessler, J. Widmer, H. Kaesemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad-hoc networks," *Ad Hoc Networks*, vol. 1, no. 4, pp. 351–369, 2003.

[19] A. Festag, P. Papadimitratos, and T. Tielert, "Design and performance of secure geocast for vehicular communication," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 5, pp. 2456–2471, Jun 2010.

[20] http://www.denso-europe.com/.

[21] http://sumo.sourceforge.net.

[22] M. Piórkowski, M. Raya, A. Lugo, P. Papadimitratos, M. Grossglauser, and J.-P. Hubaux, "Trans: realistic joint traffic and network simulator for vanets," *SIGMOBILE MC2R*, vol. 12, pp. 31–33, Jan. 2008.

[23] http://trans.epfl.ch/.

[24] http://jist.ece.cornell.edu/.

[25] "IEEE 1363a-2004, standard specifications for public-key cryptography, amendment 1: Additional techniques," 2004.

[26] K. Laberteaux and Y.-C.Hu, "Strong vanet security on a budget," in *ESCAR*, 2006.

[27] F. Kargl, E. Schoch, B. Wiedersheim, and T. Leinmüller, "Secure and Efficient Beaconing for Vehicular Networks," in *ACM VANET, short paper*, Sept. 2008.

[28] E. Schoch and F. Kargl, "On the efficiency of secure beaconing in vanets," in *ACM WiSec*, 2010, pp. 111–116.

[29] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, and A. Lioy, "On the performance of secure vehicular communication systems," *IEEE Transactions on Dependable and Secure Computing*, to appear.

[30] C. Maihöfer, "A survey on geocast routing protocols," *IEEE Communications Surveys and Tutorials*, 2nd quarter issue 2004.

[31] T. Leinmüller, C. Maihöfer, E. Schoch, and F. Kargl, "Improved Security in Geographic Ad Hoc Routing Through Autonomous Position Verification," in *ACM VANET*, 2006.