Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 12, December 2013, pg.100 – 106

RESEARCH ARTICLE

Performance Evaluation of Various Attack Detection Techniques in VANET

Noble Mary Juliet.A [1], Joan Pavithra.R[2]

[1]Department of Computer Science and Engineering, NPR College of Engineering and Technology, India

[2] Department of Computer Science and Engineering, NPR College of Engineering and Technology, India

[1]joanp19@gmail.com

Abstract— Vehicular communications play a substantial role in providing safety transportation by means of safety message exchange. Researchers have proposed several solutions for securing safety messages. Vehicular ad hoc networks aim at enhancing road safety by providing vehicle-to-vehicle communications and safety related applications. But safety-related applications, like Local Danger Warning, need a high trust level in received messages. Indeed, decisions are made depending on these messages. To increase the trustworthiness of these messages various detection techniques are used to detect the attackers in the VANET. But most of these techniques concentrate only on outsider attackers rather than insider attackers. In this research we discuss the various detection techniques such as dynamic thresholds based detection, filtering false data via authentic consensus, and efficient threshold-based event validation that are used to detect the insider attacks in VANET. Then we analyze the performance of these techniques using their simulation results.

Keywords— VANET; Dynamic-Thresholds; authentic consensus; Efficient and secure threshold-based event validation; Proof-of-Relevance

I. INTRODUCTION

Safety related applications such as cooperative collision avoidance, local danger warning and road hazard notification could save lives. In fact, alerts from these applications enable the drivers to react to dangerous situations such as obstacles or bad road conditions, hence reducing the risk of an accident. It is crucial to make sure that the life critical information in these applications cannot be forged or modified by an attacker. Vehicular networks are especially vulnerable to fake attacks where misbehaving vehicles inject erroneous information into the network to affect the behaviour of the other drivers for their selfish objectives. For example, in traffic congestion optimization, honest drivers may be misled and driven to congested area by falsely injected information, while the attacker vehicle can enjoy less traffic on its own path. More dangerously, the drivers may be misled into potential accidents.

Classification of Attackers

Attackers can be classified according to scope, nature, and behaviour of attacks [1,2]. Some types of attackers are discussed below:

1. Some attackers eavesdrop only on the wireless channel to collect traffic information which may be passed onto other attackers. As these attackers do not participate in the communication process of the network, they are called passive attackers. On the other hand, some attackers either generate packets containing wrong information or do not forward the received packets. These are called active attackers.
2. Attacker may be an authentic member of a VANET having authentic public keys and access to other members of the network. Such attackers are called insider. Outside attackers (outsider) are intruders and they can launch attacks of less diversity.
3. Some attackers are not personally benefited from the attack. Their aim is to harm other members of the network or disrupt the functionality of a VANET. These attackers are malicious. On the other hand, rational attacker seeks personal benefit and is more predictable in terms of type and target of the attack.
4. Local attacker launches an attack with a limited scope, that is, an attack is restricted to a particular area. An attack can be extended, where an attacker can control several entities distributed across the network.

II. TYPES OF INSIDER ATTACKS

Owing to the large number of autonomous network members and the presence of human factor, misbehaviour of nodes in future vehicular networks cannot be ruled out. Several types of attacks [2] have been identified and classified on the basis of layers used by the attacker. At the physical and link layer, an attacker can disturb the network system by overloading the communication channel with useless messages. An attacker can inject false messages or rebroadcast an old message also. Some attackers can tamper with an OBU or destroy an RSU. At network layer, an attacker can insert false routing messages or overload the system with routing information. Privacy of drivers can be disclosed by revealing and tracking the position of drivers. Some of these attacks are briefly explained subsequently.

A. Bogus Information

In this case, attackers are insiders, rational, and active. They can send wrong information in the network so that it can affect the behaviour of other drivers. For example, an adversary can inject wrong information about a nonexistent traffic jam or an accident diverting vehicles to other routes and freeing a route for itself.

B. Cheating with Sensor Information

This attack is launched by an attacker who is insider, rational, and active. He uses this attack to alter the perceived position, speed, and direction of other nodes in order to escape liability in case of any mishap.

C. ID Disclosure

An attacker is insider, passive, and malicious. It can monitor trajectories of a target vehicle and can use this information for determining the ID[7] of a vehicle.

D. Denial of Service (DOS)

Attacker is malicious, active, and local in this case. Attacker may want to bring down the network by sending unnecessary messages on the channel. Example of this attack includes channel jamming and injection of dummy messages.

E. Replaying and Dropping Packets

An attacker may drop legitimate packets. For example, an attacker can drop all the alert messages meant for warning vehicles proceeding toward the accident location. Similarly, an attacker can replay the packets after that event has been occurred to create the illusion of accident.

III. DYNAMIC THRESHOLDS BASED DETECTION

In this method, we are addressing the problem how to determine whether information about an event like icy road or an accident ahead is trustworthy or not. We assume that we receive information from multiple communication partners, some of which might be malicious. Applying a consensus mechanism allows the local On-Board Unit (OBU) to reach a decision before taking actions like warning the driver. Generally speaking, the OBU would require to receive a certain number of consistent reports about a specific event before a warning would be issued. Having such a consensus mechanism in place increases the trustworthiness of received

warnings at the expense of additional delay has the OBU would first have to wait for reception of a certain number of messages to reach a certain confidence threshold [5].

The chosen value will have an influence on a number of parameters like the required processing resources for checking the messages [7]. But most important, it influences the trade-off between the decision delay (and thus the delay until a driver gets warned) and the trustworthiness of the information (and thus the opportunity for an attacker to cheat). So a threshold must be chosen carefully.

As mentioned in [9], a cooperative local danger warning application comprises three steps: detection process, message dissemination, and decision process. In the detection process, vehicles detect hazards with their on-board sensors while driving. Whenever a critical condition is detected, the vehicle triggers the dissemination process and broadcasts a warning message—sent in a WSM every 100 ms [2]. Vehicles receiving such a message, trigger the decision process. If there is sufficient evidence for a critical road condition on the route ahead, the system notifies the driver to have him take appropriate actions.

We are interested in the decision process, where the LDW application has to decide whether or not to take action or notify the driver, because leading the system into a wrong decision is one of the major threats. Kalman filter is used to make the decision process, it is a set of mathematical equations that provides a effective means of estimating the states of a process. Notations used in method are,

-$T_{collision}$: the expected collision time computed by the speed and the distance.

- $T_{braking}$: the time of braking computed by $T_{braking} = v_k a$, where a is the deceleration rate and v_k the speed of the vehicle V_k.

- $T_{reaction}$: the reaction time of the driver (0.7-1.5 second).

- T_{safety}: the time to travels the safety distance, which is computed by $T_{safety} = T_{braking} + T_{reaction}$.

- ΔT_i: the remaining time before the collision, which is computed by $\Delta T_i = T_{collision} - t_i$ (where t_i is the current time).

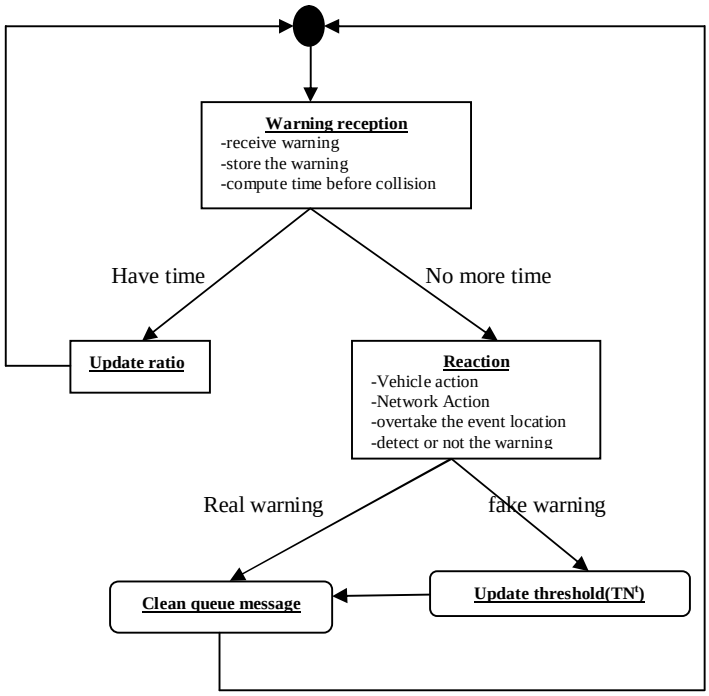


Fig.1. state machine of threshold determination

A.Threshold determination:-

The threshold determination is defined as a cyclic state machine (cf. Fig.1). When a vehicle receives a warning, it computes ΔT_i. If $\Delta T_i > T_{safety}$, then the remaining time permits to collect more messages. The ratio is updated. Otherwise, there is no more time to collect and the vehicle has to make a decision with the current set

of messages received. When the vehicle overpasses the danger location and does not detect a danger, it updates the threshold TN^t. Hence the vehicle becomes more suspicious. Whatever the result of the detection, the vehicle cleans its queue message by deleting the warnings that have a coordinate behind it.

IV. DETECTION OF FALSE DATA VIA AUTHENTIC CONSENSUS

This method presents the notion of Proof-of-Relevance (PoR), which consists in proving that the event reporter is authentically relevant to the event it has reported. The PoR is accomplished by collecting authentic consensus on the event from witness vehicles in a cooperative way. Event reports from attackers who fail to provide this PoR are disregarded, making the network immune to bogus data. The general notion of PoR can be implemented in various ways. In particular, PoR can be achieved via authentic consensus, constituted by the vehicles collecting digital endorsements from other witnesses in the detecting area. After collecting a number of endorsements to reach a verifiable consensus, vehicles disseminate the information along their routes to notify other drivers. On receiving the information, vehicles can accept and respond only after they have verified all the signatures in the event report. In this way, PoR keeps the network immune to bogus data. Second, it should be noticed that an efficient and secure signature collection protocol is a key component to reach such an authentic consensus.

A. Report Generation

Once a vehicle detects some event, it generates an event report of the following format $E = \{LE,D, t\}$, where LE is the location of the event, D is the event type and t is the event time.

B. Signature Collection using Growth Code

Signature collection is a key procedure in our Proof-of-Relevance scheme. Vehicles detecting the event will participate in the signature collection protocol on the same event until they collect more than T signatures. To avoid the duplicate transmissions growth code is used. Growth Code [10] is a kind of erasure code whose degree grows with time initially proposed to enhance data persistence in sensor networks.

Fig.2. depicts the working of signature collection using growth codes. Initially vehicles send and receive low degree code words which are able to be decoded immediately. Later on, they send and receive codewords with higher degrees which are more valuable than lower degree codewords in recovering more symbols. Second, since Growth Code can be decoded with good opportunities, the receiver can check the newly decoded symbol [1] by verifying the correctness of the signature.

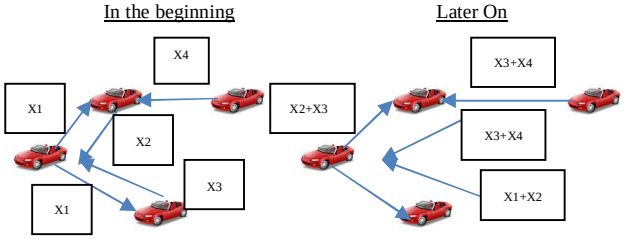


Fig.2. Signature Collection using Growth Code

C. Report Verification and Decision

When a node receives a report, it first examines whether there are enough signatures in the report. Reports with less than T signatures will be discarded. If there are T signatures, the node goes on to validate each signature in the report using the corresponding public key. If any of the T signatures is incorrect, the packet will be discarded. If all the T signatures are checked as valid, the vehicle will accept the message and react according to the event type.

V. THRESHOLD-BASED EVENT VALIDATION

Determining whether the number of vehicles reporting an event is above a threshold is an important mechanism for VANETs, because many applications rely on a threshold number of notifications to reach agreement among vehicles, to determine the validity of an event, or to prevent the abuse of emergency alarms. We present the first efficient and secure threshold-based event validation protocol for VANETs.

To obtain high certainty for a MH(Multi-Hop)-relevant event, vehicles rely on a threshold number of vehicles to report that event before alerting the driver. The core challenge in threshold based event validation for

VANETs is to create an efficient mechanism to combine and distribute event alerts with a low error rate in the presence of malicious entities.

Collection phase: Once a vehicle observes an event, that vehicle begins broadcasting alerts about the event and starts to collect other vehicles' alerts pertaining to the event. Specifically, a witness vehicle broadcasts a triple (□, σ, cert), where □ is an event description, σ is a signature on □, and cert is a public-key certificate. To reduce communication overhead in the Collection phase, a witness only keeps a synopsis, a subset of alerts providing a rough estimate of number of alerts (ñ). The witness vehicles exchange synopses with each other using the Message Exchange Protocol. The Collection phase is finished when the threshold-based validation algorithm determines that the vehicle has collected sufficient alerts to generate an event proof (a synopsis showing $\tilde{n} \geq \tau$), or when the event expires. If $\tilde{n} \geq \tau$, the witnesses transit to the Distribution phase to spread the snopsis.

Distribution phase: After receiving an event proof that indicates $n \geq \tau$, vehicles rebroadcast the event proof to alert vehicles further away. Similar to in the Collection phase, in the Distribution phase, the rebroadcast frequency and message payload is determined by the message exchange protocol. By verifying an event proof, a vehicle away from the event scene can be assured that the total number of alerts exceeds a certain threshold value ($n \geq \tau$) without hearing all of the n alerts.

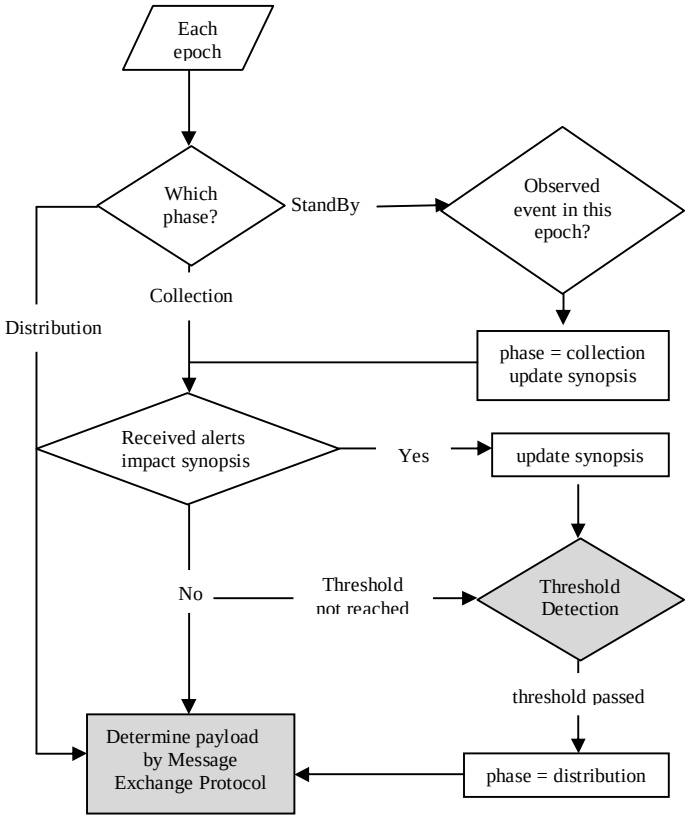


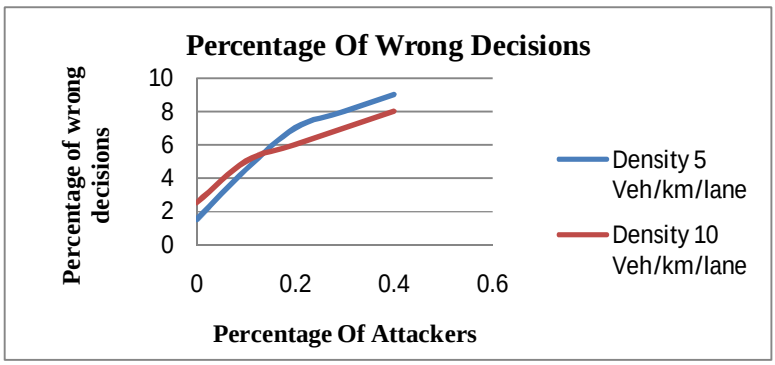
Fig.3.The phase transitions and operations in threshold-based applications.

Fig.3. outlines the phase transitions in threshold-based applications. During the Standby phase, there is no active MH-relevant event. In the occurrence of multiple concurrent events, the applications maintain per-event phase and synopsis, but broadcast their synopses in the same beacon.

VI.PERFORMANCE EVALUATION

Dynamic threshold-based detection is used to increase trust in local danger warning by detecting spoofed data. More specifically, we modelled the threshold as a Kalman filter. We proposed an algorithm similar to a learning scheme to dynamically adjust this threshold. Thus, the threshold estimates the current percentage of attackers in the VANET. We analyzed the impact of the density and the percentage of attackers on the decision delay and the percentage of wrong decisions. This method overestimates the presence of attackers but leads to protect vehicle from spoofed data injection. we conclude that the default threshold value should be

chosen carefully to shorten the inevitable bootstrapping phase. Future work on this method is to use the extensive simulations to assess the delay to achieve a best-suited threshold.



From the Fig.4.we observe that the simulation time needs to be increased to analyze the delay needed to reach a stable threshold (because this delay is strongly dependent of the scenario), and to increase the number of suffered events per vehicle.

In filtering false data via authentic consensus, we have considered two different scenarios for the signature collection protocol. The first one is the dense scenario with man vehicles as detectors, for example in traffic jams. The second one is the sparse scenario with much fewer detectors, such as a road hazard happening in the early morning.

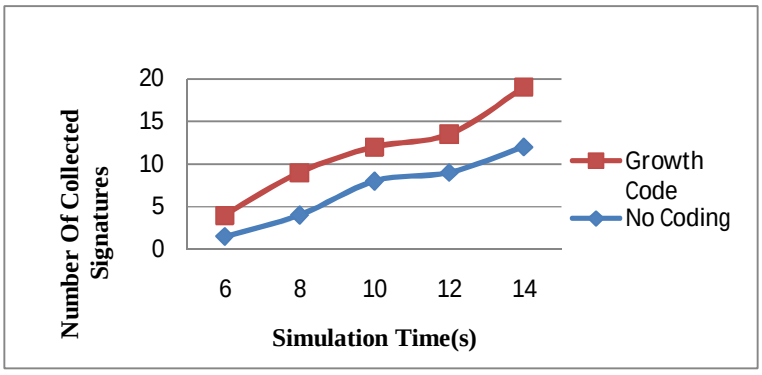


Fig.5 Average Number Of Collected Signatures Vs Simulation Time in the Sparse Scenario

Efficiency of signature collection: Our signature collection protocol outperforms simple flooding protocol. Fig.2 and Fig.3 depict the average number of signatures collected versus the elapsed time in our simulation in sparse and dense scenarios respectively, which demonstrates around 10% and 50% enhancements respectively. This scheme also indicate the average number of collected signatures versus the average number of codewords in sparse and dense scenarios respectively, in which we can see the signature collection using growth codes could help the vehicles collect more signatures with fewer transmissions.

NS-2 simulator is used to measure the impact of threshold-based validation algorithms and message exchange protocol (MEP) on network performance and the delay associated with distributing an event proof. The results of simulation show that the MEP protocol, which rebroadcasts synopses intelligently, can distribute a proof of congestion to vehicles 4.5 kilometers away from the congestion area in less than 1 second with little impact on network performance. The vehicles are represented as mobile nodes in the simulation. Every 0.1 seconds an vehicle sends out a beacon that contains the safety information and any MH-relevant application data.

VII. CONCLUSION

The design of security solution in vehicular ad hoc networks attracts more and more attention from research groups. Indeed VANETs are extremely vulnerable to attacks, due their shared wireless medium and the absence of conventional security infrastructures. In this article we describe the different types of attacks, attackers and the attacker detection schemes such as dynamic threshold-based detection, filtering false data via

authentic consensus and efficient threshold-based event validations. The performance analysis of these techniques shows that they are effectively detect the attacks, however there are many challenges need to be faced with these techniques. In future work these techniques need to be evaluated in realistic manner with some enhancements.

REFERENCES

[1] M. Raya and J.-P. Hubaux,(2007)" *Securing vehicular ad hoc networks*". Journal of Computer Security, 15(1), 39–68.

[2] A. Aijaz, B. Bochow, F. Dtzer, A. Festag, M.Gerlach, R. Kroh, andT. Leinmuller,(2006) "*Attacks on inter-vehicle communication systems—an analysis*". In Proceedings of the 3rd international Workshop on Intelligent Transportation (WIT).

[3] J. Petit, M. Feiri, and F. Kargl, "*Spoofed data detection in VANETs using dynamic thresholds*," in *Proc. IEEE VNC*, Nov. 2011, pp. 25–32.

[4] H.-C. Hsiao, A. Studer, R. Dubey, E. Shi, and A. Perrig, "*Efficient and secure threshold-based event validation for VANETs*," in *Proc. 4th ACM Conf. WiSec*, New York, 2011, pp. 163–174.

[5] Z. Cao, J. Kong, U. Lee, M. Gerla, and Z. Chen, "*Proof-of-relevance: Filtering false data via authentic consensus in vehicle ad-hoc networks*," in *Proc. IEEE INFOCOM*, 2008, pp. 1–6.

[6] 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 Commun. Mag.*, vol. 46, no. 11, pp. 100–109, Nov. 2008.

[7] J. Petit, "*Analysis of ecdsa authentication processing in vanets*," in *Proceedings of the 3rd international conference on New technologies, mobility and security (NTMS'09)*, Cairo, Egypt, 2009, pp. 388–392.

[8] X. Sun, X. Lin, and P. Ho, (2007) " *Secure vehicular communications based on group signature and ID-based signature scheme"*.In Proceedings of the IEEE International Conference on Communications.

[9] B. Ostermaier, F. D¨otzer, and M. Strassberger, "*Enhancing the security of local danger warnings in vanets - a simulative analysis of voting schemes*," in *Proceedings of the 2nd International Conference on Availability, Reliability and Security (ARES'07)*, Vienna, Austria, April 2007, pp. 422–431.

[10] Abhinav Kamra, Jon Feldman, Vishal Misra, and Dan Rubenstein, "*Growth codes: Maximizing sensor network data persistence*," in *Proceedings of ACM Sigcomm*, Pisa, Italy, September 2006.

