Detecting Sybil Attacks using Proofs of Work and Location in VANETs

Butukuru Rojalakshmi, Vanapamula Veerabrahmachari, Chevula Rekha, Arekatla Jaganmohan Reddy

[1,3] Assistant Professor, [2,4] Associate Professor

brojalakshmi@gmail.com, vveerabrahmachari@gmail.com

rekhavenkat16@gmail.com, jagan.arekatla@gmail.com

Department of CSE, A.M. Reddy Memorial College of Engineering and Technology, Petlurivaripalem,

Narasaraopet, Andhra Pradesh -522601

ABSTRACT

Intelligent Transportation Systems (ITS) of the future may be possible with the help of Vehicular Ad Hoc Networks (VANETs). With the use of ITS, data collected from cars may provide a spatial-temporal picture of traffic statistics, leading to better road safety and less congestion. Vehicles should not be identified by a single identity but rather by a combination of pseudonyms in order to protect their anonymity. But cars may take advantage of all the pseudonyms out there and conduct Sybil assaults by impersonating other vehicles. Then, these Sybil--er, fictitious vehicles—report inaccurate information in order to—gasp!—inflate traffic or skew traffic management metrics. A Sybil attack detection strategy using proofs of work and location is proposed in this research. The basic premise is that every roadside unit (RSU) will provide a signed, time-stamped tag to verify the vehicle's anonymous whereabouts. The vehicle's anonymous identity is derived from its trajectory, which is created using proofs transmitted by many successive RSUs. Furthermore, several RSUs are required to give vehicle trajectories; a single RSU cannot do so. This manner, attackers can't generate phony trajectories without compromising an unrealistic amount of RSUs. In addition, when an RSU has provided the evidence of location, the car should use the proof of work (PoW) technique to solve a computational challenge. Therefore, before it can get a proof of location, it has to provide the next RSU a valid solution, also called proof of work.

In the scenario of low-density RSUs, the vehicles may be prevented from generating different trajectories by using the PoW. The event manager then employs a matching approach to detect the trajectories sent by Sybil cars in the midst of any recorded occurrence, such as traffic jams. The plan hinges on the paths of the Sybils being physically tied to a single vehicle, which means that they will overlap. Simulations and testing show that our approach successfully detects Sybil assaults with a high detection rate, low false negative rate, and reasonable computation and communication cost.

.

INTRODUCTION

A foundational component of the next generation of Intelligent Transportation Systems (ITSs), Vehicular Ad Hoc Networks (VANETs) have been making roadways safer and more efficient over the last 20 years. Within VANETs, cars in motion are able to communicate not only with one another but also with nearby roadside units (RSUs) via RSU-to-vehicle communications. Consequently, several applications have surfaced as potential answers, opening the door to new kinds of omnipresent traffic control applications that our present-day conventional transportation system cannot support. The main concept behind these apps is to let cars provide data and comments to an event manager, who can then use that information to create a spatial-temporal picture of the traffic situation and extract useful congestion statistics [2]. By facilitating a variety of applications, such as better route guidance and navigation, local hazard notification, traffic flow management, pre-crash sensing and warning, and more, these technologies may help make roads safer and more efficient [3].

Nevertheless, data sent by participating automobiles is essential for the aforementioned applications. Consequently, it is necessary to anonymously verify drivers' identities while protecting their privacy, particularly their location privacy [4, 5]. As a simplistic fix, we can just let each car have its own set of pseudonyms for anonymous authentication. The privacy feature might be used by an evil vehicle to undertake a Sybil assault, however [6]. When launching a Sybil assault, a malicious vehicle will utilize its pseudonyms to pose as several Sybil nodes [7]. A Sybil assault on VANETs might have catastrophic results. To make it seem like there's traffic jam, a malevolent car, for instance, may commence the assault. Because of this, other drivers will forego the road in order to avoid the dangerous car. In safety-related applications like danger alerts and collision avoidance systems, a Sybil attack might provide skewed findings, which could lead to accidents [3]. As a result, finding Sybil attacks in VANETs is crucial.

There are now three main schools of thought when it comes to identifying Sybil attacks: identity registration, position verification, and trajectory-based techniques.

Making ensuring every physical node has a distinct and legitimate identification is the end aim of these detection systems. The first step in identity registration methods [7-9] is to set up a specific vehicular public key infrastructure

that can certify cars using many aliases. This will make sure that every physical node has a distinct and legitimate identity. Unfortunately, Sybil attacks may still happen even with identity registration in place. This is because a bad node can acquire several identities by non-technical ways, such theft or even vehicular collusion [10]. Second, the idea that a single vehicle may only be in one place at any given moment is foundational to position verification methods. The use of global positioning systems (GPS) and other localization methods to give vehicle location information for Sybil node detection is discussed in references [11] and [3]. Since vehicle networks are inherently dynamic, these strategies fail [12]. Thirdly, the idea that cars are autonomous and should, therefore, follow separate paths is central to trajectory-based systems. According to [4], the vehicle gets its path by adding up the tags it receives from RSUs in a sequential manner. Unfortunately, the approach is vulnerable to RSU compromise attacks, where a hostile vehicle may get an endless number of legitimate trajectories by compromising only one RSU.
In addition, because RSUs are not prevalent in rural regions, attackers may construct legitimate itineraries to search for various vehicles. In order to identify Sybil trajectories, when other cars provide events to the event manager, it utilizes a set of heuristics to build a network of Sybil nodes. Then, it employs the maximum clique method [14] to identify every Sybil node in that graph.
Here are the key points of our work and the problems that this article seeks to solve:
The use of threshold signatures allowed us to withstand RSU hacking attempts. In order to generate false trajectories, the attacker must give over an impossible amount of RSUs. _ Using the POW method, we were able to decrease the detection time for Sybil trajectories—a major issue in traffic management applications—and to restrict the capacity of a rogue vehicle to construct several forged trajectories.
_ By experimentally studying the influencing parameters (e.g., travel time between two consecutive RSUs), we thoroughly examined the probabilistic nature of the POW-based scheme. Based on this analysis, we developed a mathematical model that can be used to adjust these parameters, thereby significantly reducing the ability of a malicious vehicle to create forged trajectories. Through our tests, we have shown that the proof of work method hinders the malevolent vehicle's capability to concurrently maintain numerous trajectories. Results show that the proposed scheme can detect and defend against Sybil attacks in VANETs more efficiently than the Footprint, and further simulations, analysis, and practical experiments are carried out to evaluate it [4].

Exisiting System

A certificate-based privacy-preserving technique for Sybil node detection was suggested by Zhou et al. [8]. Vehicles are provided with a pool of pseudonyms to mask their unique identification by the department of motor vehicles (DMV), which represents the certificate authority. A common value is generated by hashing the pseudonyms linked to each vehicle. An RSU can tell whether the received pseudonyms are from the same pool by computing their hashed values. RSUs have the ability to identify Sybil nodes and notify the DMV of any suspicious automobiles.
In order to protect RSUs against compromise, the study proposes using coarse-grained keys and fine-grained keys in two-level hash functions. Since RSU only stores valid coarse-grained keys for a limited period of time, it is unable to determine whether the pseudonyms are associated with a specific vehicle. While DMV keeps all keys and can identify Sybil nodes using two-level hashing, an attacker compromising an RSU only receives the coarse-grained hash key for the current time period. While using trustworthy certificates is the best way to stop Sybil assaults in their tracks, it compromises entities' ability to remain anonymous and their ability to control where they are located.
Another issue with large-scale networks like VANETs is the need on a centralized authority to provide unique identities to each node. A method for anonymous group authentication was suggested by Chen et al. in [30] and it is based on group signatures. At the same time, the verifier can tell whether the same node has signed the same message more than once. The elimination of Sybil attacks may be achieved, thus, by recognizing signatures signed by the same cars. On the other hand, if the malicious vehicle can come up with messages that seem similar, he may initiate Sybil attacks. To create the trust connection among the participating entities, Reddy et al. [7] has suggested a mechanism based on cryptographic digital signatures.
To guarantee confidence amongst participating nodes, the most applicable method to our study is to use vehicle trajectories as its IDs. Digital signatures with a timestamp are sent to cars under the coverage of RSUs in [32]. As they traveled, vehicles collected signatures from RSUs and stored them. But since the time stamp isn't associated with a specific vehicle, an adversarial vehicle might theoretically eavesdrop on a wireless channel and assert its presence at a certain RSU, even if it wasn't really there. In order to identify Sybil attacks, Footprint was developed in [4]. A vehicle may prove its existence at a certain place and time by obtaining a signed message whenever it passes an RSU. The succession of permitted communications that a vehicle collects as it continues to move makes

up its trajectory. The similarity in the trajectories produced by an attacker makes a Sybil assault easy to spot. But there are several major problems with Footprint.
The lack of implementation of hash keys in the system makes it unable to detect Sybil assaults. In order to withstand Sybil and DDOS assaults, the system does not have attack resistance measures deployed.

Suggested Framework
Here, we provide a new method for detecting Sybil attacks that makes use of location and proofs of work. The basic concept is that RSUs should concatenate the time of appearance and anonymous location tags of themselves and give permitted time-stamped tags to vehicles whenever they come into contact with them. The vehicle builds its trajectory as it continues to move by merging a series of successive permitted time-stamped tags that are sequentially linked to one another. The vehicle's anonymous identity is derived from that trajectory. We build the trajectory so that several RSUs may generate trajectories for the cars, as RSUs are primarily responsible for providing evidence of position to them. This way, the method can withstand attacks that compromise RSUs. This is accomplished by implementing threshold signature, which limits each RSU to producing a partial signature on a set of time-stamped tags.
A standard signature that serves as evidence of location may be created as a vehicle travels along a certain threshold number of RSUs. The vehicle is expected to utilize an allowed message as a seed to solve a problem using a proof-of-work mechanism, similar to Bitcoin's, when it receives one from an RSU [13]. Provide a proof to RSUs so they can confirm the vehicle solved the challenge successfully; this is the essential principle of PoW. The capacity of malevolent vehicles to generate various paths is reduced while using PoW, in comparison to Footprint [4].
Upon receiving an event from other cars, the event manager utilizes a set of heuristics to form a linked network of Sybil nodes. Then, it employs the maximum clique method [14] to find all Sybil nodes in that graph, which allows it to detect Sybil trajectories. The Benefits Protecting against RSU compromise attacks, the system made use of threshold signatures. In order to generate false trajectories, the attacker must compromise an impossible amount of RSUs. _ The system utilized the PoW algorithm in conjunction with machine learning classifiers to decrease the detection time for Sybil trajectories—a crucial concern in traffic management applications—and to limit the ability of a malicious vehicle to create multiple forged trajectories. By experimentally examining the affecting parameters (such as the travel time between two consecutive RSUs), the system thoroughly analyzed the probabilistic nature of the PoW based scheme. We then developed a mathematical model that can be used to adjust these parameters, reducing the ability of a malicious vehicle to create forged trajectories. Through our tests, we have shown that the proof of work method hinders the malevolent vehicle's capability to concurrently maintain numerous trajectories. After running additional simulations, analyses, and experiments to compare the suggested scheme to the Footprint [4], the results show that the proposed scheme is more efficient than the Footprint and can detect and defend against Sybil attacks in VANETs.

Logistic regression Classifiers

Logistic regression analysis examines the relationship between a group of independent factors that might be used to explain a categorical dependent variable. When there are only two possible values for the dependent variable, like yes or no, logistic regression is used. When the dependent variable may take on three or more distinct values, such "Married," "Single," "Divorced," or "Widowed," the method is known as multinomial logistic regression.
Despite a difference in the dependent variable data format compared to multiple regression, the procedure's practical application is comparable. As an alternative to discriminant analysis, logistic regression may be used to examine categorical response variables. When compared to discriminant analysis, logistic regression is seen by many statisticians as being more flexible and appropriate for modeling a wider range of scenarios. Reason being, unlike discriminant analysis, logistic regression does not presume regularly distributed independent variables.
Using both numerical and categorical independent variables, this application calculates multinomial logistic regression and binary logistic regression. Equation for regression, goodness-of-fit, odds ratios, confidence intervals, probability, and deviation are all reported. It generates diagnostic residual reports and graphs as part of its thorough residual analysis. It is capable of searching for the optimal regression model using the minimum number of independent variables using an independent variable subset selection search. It helps find the optimal classification cutoff point by providing ROC curves and confidence intervals on predicted values. By automatically categorizing rows that aren't utilized in the study, it lets you verify your findings.
Simple Bayes
As a supervised learning technique, the naive bayes approach takes the oversimplified premise that one class characteristic's existence or absence has no effect on the presence or absence of any other feature.

However, this does not diminish its seeming robustness and efficiency. When compared to other supervised learning methods, its performance is on par. A number of arguments have been put forward in the published works. We emphasize a representation bias-based explanation in this lesson. Along with linear discriminant analysis, logistic regression, and linear SVM, the naive bayes classifier is a linear classifier (support vector machine). How the classifier's parameters are estimated (the learning bias) is where the difference is.
The Naive Bayes classifier has a large user base in academia, but it's not that popular among practitioners who are looking for practical results. Among its many advantages, academics have noted that it is simple to code and put into practice, has easily estimable parameters, learns quickly even on massive datasets, and outperforms competing methods in terms of accuracy. However, end users do not get a model that is simple to comprehend and implement, and they fail to grasp the value of this approach. As a consequence, we incorporate the findings of the training into a fresh presentation. The classifier is more intuitive, and it's simpler to implement. Part one of this guide covers the fundamentals of the naive bayes classifier from a theoretical standpoint. We then apply the method to a dataset using Tanagra. We evaluate the model parameters and compare them to results from logistic regression, linear discriminant analysis, and linear support vector machines, among other linear methods. The findings are quite consistent, which is something we noticed. For the most part, this explains why the technique outperforms alternatives. Weka 3.6.0, R 2.9.2, Knime 2.1.1, Orange 2.0b, and RapidMiner 4.6.0 are some of the tools used on the same dataset in the second portion. Above all else, we strive to comprehend the outcomes.

Random Forest

An ensemble learning technique for classification, regression, and other problems, random forests (sometimes called random decision forests) work by building a large number of decision trees during training. The majority tree selection is the result of a random forest when it comes to classification jobs. In regression tasks, the average or mean prediction from each tree is given back. The tendency of decision trees to overfit their training set may be mitigated by using random decision forests. While random forests do better than choice trees in most cases, they are not as accurate as gradient enhanced trees. Their efficiency is, however, susceptible to data quality factors.

Tin Kam Ho[1] developed the first random decision forest algorithm in 1995 utilizing the random subspace technique. This method, according to Ho's formulation, is a means to execute Eugene Kleinberg's "stochastic discrimination" approach to classification.

Minitab, Inc. now owns the trademark "Random Forests" that was filed in 2006 by Leo Breiman and Adele Cutler, who created an algorithm extension.In order to build a set of decision trees with controlled variance, the expansion incorporates Breiman's "bagging" concept with random feature selection, which was first proposed by Ho[1] and subsequently separately by Amit and Geman [13].

Businesses often use random forests as "blackbox" models because they provide good predictions on a variety of data sets with little setup.

SVM

An iid training dataset is used by discriminant machine learning techniques to create a discriminant function that can accurately predict labels for newly acquired instances in classification tasks.

A discriminant classification function takes a data point x and assigns it to one of the classes that are part of the classification job, in contrast to generative machine learning techniques that need calculations of conditional probability distributions. Although discriminant methods aren't as effective as generative ones—the latter are often used for outlier identification in predictions—they do utilize less training data and computer resources, which is great for a multidimensional feature space and for situations where just posterior probabilities are required.

Learning a classifier is geometrically similar to discovering the equation of a multidimensional surface that optimally divides the feature space into its respective classes.

In contrast to perceptrons and genetic algorithms (GAs), which are often employed in machine learning for classification, support vector machines (SVMs) always yield the same ideal hyperplane value due to their analytical solution of the convex optimization issue. Both the starting and ending points of a perceptron have a significant impact on the results.

The parameters of the SVM model are uniquely determined for a certain training set and a specific kernel that maps the input space to the feature space; in contrast, the perceptron and GA classifier models are changed with each training iteration. Since the only purpose of GAs and perceptrons is to reduce training errors, this criterion will be satisfied by a number of hyperplanes.

Results:

CONCLUSION

In VANETs, Sybil assaults may have devastating effects. In this research, we provide a new method for identifying Sybil assaults by combining location and proofs of work. In VANETS, Sybil attacks may have devastating effects since they construct an anonymous vehicle trajectory by collecting a series of evidence of locations from several RSUs. Using location and proofs of work, we provide a new method for identifying Sybil assaults in this study. By collecting a series of proofs of locations from various RSUs that a vehicle contacts, an anonymous trajectory may be built. To prevent an RSU compromise attack, it is necessary to have a minimum of t RSUs in order to generate a proof-of-location message using a threshold signature, rather than enabling a single RSU to send approved messages to cars. Furthermore, proof-of-work algorithms may restrict malevolent vehicles' capacity to generate fake trajectories. The results of our tests show that our system is capable of detecting Sybil assaults with a high sensitivity and a low false negative rate. The packets that are transferred have an appropriate amount of communication and processing overhead.

REFERENCES

[1] F.-J. Wu and H. B. Lim, "Urbanmobilitysense: A user-centric participatory sensing system for transportation activity surveys," IEEE Sensors Journal, vol. 14, no. 12, pp. 4165–4174, 2014.

[2] S. Hu, L. Su, H. Liu, H. Wang, and T. F. Abdelzaher, "Smartroad: Smartphone-based crowd sensing for traffic regulator detection and identification," ACM Transactions on Sensor Networks (TOSN), vol. 11, no. 4, p. 55, 2015.

[3] K. Rabieh, M. M. Mahmoud, T. N. Guo, and M. Younis, "Cross-layer scheme for detecting large-scale colluding sybil attack in vanets," in 2015 IEEE International Conference on Communications (ICC). IEEE, 2015, pp. 7298–7303.

[4] S. Chang, Y. Qi, H. Zhu, J. Zhao, and X. Shen, "Footprint: Detecting sybil attacks in urban vehicular networks," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 6, pp. 1103–1114, 2012.

[5] Z. MacHardy, A. Khan, K. Obana, and S. Iwashina, "V2x access technologies: Regulation, research, and remaining challenges," IEEE Communications Surveys & Tutorials, vol. 20, no. 3, pp. 1858–1877, 2018.

[6] F. Qu, Z. Wu, F.-Y. Wang, and W. Cho, "A security and privacy review of vanets," IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 6, pp. 2985–2996, 2015.

[7] D. S. Reddy, V. Bapuji, A. Govardhan, and S. Sarma, "Sybil attack detection technique using session key certificate in vehicular ad hoc networks," in Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), 2017 International Conference on. IEEE, 2017, pp. 1–5.

[8] T. Zhou, R. R. Choudhury, P. Ning, and K. Chakrabarty, "P2dapsybil attacks detection in vehicular ad hoc networks," IEEE journal on selected areas in communications, vol. 29, no. 3, pp. 582–594, 2011.

[9] K. El Defrawy and G. Tsudik, "Privacypreserving location-based ondemand routing in manets," IEEE journal on selected areas in communications, vol. 29, no. 10, pp. 1926–1934, 2011.
[10] Y. Yao, B. Xiao, G. Wu, X. Liu, Z. Yu, K. Zhang, and X. Zhou, "Multichannel based sybil attack detection in vehicular ad hoc networks using rssi," IEEE Transactions on Mobile Computing, 2018.