Leveraging Proofs of Work and Location-Based Strategies for Sybil Attack Detection in VANETs

***M IMADAD ALI**, Assistant Professor*
Department of MCA
Santhiram Engineering College, Nandyal, A.P

ABSTRACT

Vehicular Ad Hoc Networks (VANETs) have the capability to facilitate the development of the next-generation Intelligent Transportation Systems (ITS). Through the use of Intelligent Transportation Systems (ITS), the data collected from vehicles may be utilized to construct a comprehensive understanding of traffic patterns over space and time. This information can then be leveraged to enhance road safety and mitigate congestion and traffic jams. In order to safeguard the privacy of cars, it is advisable for them to utilize various pseudonyms rather than relying on a single identity. Nevertheless, vehicles can take advantage of the wide supply of pseudonyms and carry out Sybil assaults by impersonating several vehicles. Subsequently, these Sybil (or counterfeit) vehicles transmit inaccurate information, such as fabricating traffic congestion or contaminating traffic control data. This paper presents a Sybil assault detection system that utilizes proofs of work and location. The concept entails each road side unit (RSU) generating a signed time-stamped tag to serve as evidence of the vehicle's unidentified position. The vehicle trajectory, which serves as the vehicle's anonymous identity, is generated by utilizing proofs transmitted from a series of consecutive RSUs. In addition, a single RSU lacks the capability to generate vehicle trajectories. Instead, the combined efforts of multiple RSUs are required. Attackers would need to compromise an impractical number of RSUs in order to construct counterfeit trajectories. In addition, once the vehicle receives the proof of position from an RSU, it must solve a computational challenge by executing a proof of work (PoW) algorithm. In order to get a proof of location, the device must first give a valid solution (proof of work) to the next RSU. Employing the Proof of Work (PoW) mechanism can effectively mitigate the issue of vehicles generating different routes while encountering low-density Roadside Units (RSUs). During any reported event, such as road congestion, the event manager employs a matching algorithm to determine the trajectories transmitted by Sybil cars. The technique relies on the premise that the Sybil trajectories are physically confined to a single vehicle, thereby necessitating their paths to intersect. Our method has been extensively tested through experiments and simulations, and it has been proven to produce a high detection rate for Sybil assaults. Additionally, it has a low false negative rate and acceptable levels of communication and computing overhead.

I. INTRODUCTION

Vehicular Ad Hoc Networks (VANETs) have been more important in the past twenty years as a fundamental component of the future Intelligent Transportation Systems (ITSs). They play a crucial role in enhancing road safety and improving traffic efficiency. In VANETs, mobile vehicles are capable of intercommunicating with each other through intervehicle communications, as well as with nearby road-side units (RSUs) through RSU-to-vehicle communications. Consequently, a diverse range of applications has emerged as prospective solutions [1] to facilitate novel forms of ubiquitous traffic control applications that are not feasible with our current conventional transportation system. The primary concept behind these apps is to allow vehicles to provide data and input to an event manager, who can then create a spatial and temporal representation of the traffic conditions and extract significant congestion statistics [2]. These applications can enhance road safety and efficiency by providing many functions, including pre-crash sensing and warning, traffic flow control, local danger notification, and improved route guiding and navigation. [3].

However, the aforementioned applications depend on information sent from participating vehicles. Therefore, it is required to preserve drivers privacy especially location privacy while still verifying their identities in an anonymous manner [4], [5]. A naive solution is to allow each vehicle to have a list of pseudonyms to be authenticated anonymously. However, a malicious vehicle may abuse this privacy protection to launch Sybil attack [6]. In Sybil attacks, a malicious vehicle uses its pseudonyms to pretend as multiple fake (or Sybil) nodes [7]. The consequences of a Sybil attack in VANETs can be disastrous. For example, a malicious vehicle can launch the attack to create an illusion of traffic congestion. Consequently, other vehicles will choose an alternative route and evacuate the road for the malicious vehicle. Another potential consequence of a Sybil attack is in safety-related applications such as collision avoidance and hazard warnings where a Sybil attack can lead to biased results that may result in car accidents [3]. Hence, it is of great importance to detect Sybil attacks in VANETs.

Existing works of detecting Sybil attacks can be categorized into three categories, namely, identity registration, position verification and trajectory-based approaches. The ultimate goal of these detection mechanisms is to ensure each physical node is bounded with a valid unique identity. Firstly, identity registration approaches [7–9] require a dedicated vehicular public key infrastructure to certify individual vehicles with multiple pseudonyms to ensure each physical node is bounded with a valid unique identity. However, identity registration alone cannot prevent Sybil attacks, because a malicious node may get multiple identities by non-technical means such as stealing or even collusion between vehicles [10]. Secondly, position verification approaches depend on the fact that individual vehicle can present at only one location at a time. In [11], [3], localization techniques such as Global Positioning System (GPS) are used to provide location information of vehicles to detect Sybil nodes. However, these schemes fail due to the highly mobile context of vehicular networks [12]. Thirdly, trajectory-based approaches is based on the fact that individual vehicles move independently, and therefore they should travel along different routes. In [4], the vehicle obtains its trajectory by combining a consecutive tags from RSUs which it encounters. However, the scheme suffer RSU compromise attack in which if one RSU is compromised, a malicious vehicle can obtain infinite number of valid trajectories. Moreover, in case of rural areas (RSUs are not dense), attackers can create valid trajectories that look for different vehicles.

In this paper, we propose a novel Sybil attack detection scheme using proofs of work and location. The main idea is that when a vehicle encounters an RSU, the RSU should issue authorized time-stamped tag which is a concatenation of time of appearance and anonymous location tag of that RSU. As the vehicle keeps moving, it creates its trajectory by combining a set of consecutive authorized time-stamped tags that are chronologically chained to each other. That trajectory is used as an anonymous identity of the vehicle. Since RSUs have the main responsibility to issue proof of location to vehicles, the scheme should resist against RSU compromise attack so we design the trajectory so that not only one RSU is capable of creating trajectories for the vehicles. To achieve this, threshold signature is adopted so that each

RSU is only able to generate a partial signature on a set of time-stamped tags. Once a vehicle travels along a certain threshold number of RSUs, a standard signature representing a proof of location can be generated. Upon receiving an authorized message from an RSU, the vehicle should use it as a seed to solve a puzzle using a proof-of-work algorithm, similar to the one used in Bitcoin [13]. The core idea of POW is to provide a proof to RSUs so they can ensure that the vehicle solved the puzzle correctly. Comparing to Footprint [4], using POW limits the ability of a malicious vehicles to create multiple trajectories.

To detect Sybil trajectories, upon receiving an event from other vehicles, the event manager first applies a set of heuristics to construct a connected graph of Sybil nodes, then it uses the maximum clique algorithm [14] to detect all Sybil nodes in that graph.

Our main contributions and the challenges the paper aims to address can be summarized as follows:

_ We used threshold signatures to resist RSU compromise attacks. The attacker needs to compromise an infeasible number of RSUs to be able to create fake trajectories.

_ We used the POW algorithm to limit the ability of a malicious vehicle to create multiple forged trajectories, and more importantly, to reduce the detection time for detecting Sybil trajectories which is a critical concern in traffic management applications.

_ We carefully analyzed the probabilistic nature of POW based scheme by examining the affecting parameters (e.g travel time between two consecutive RSUs) experimentally, and then we developed a mathematical model that can be used for adjusting these parameters so that the ability of a malicious vehicle to create forged trajectories is reduced significantly.

_ By experiments, we prove that using the proof of work algorithm reduces the ability of a malicious vehicle to maintain actual multiple trajectories simultaneously. Further simulations, analysis, and practical experiments are conducted to evaluate the proposed scheme and compare it with the Footprint [4], the results indicate that the proposed scheme can successfully detect and defend against Sybil attacks in VANETs and more efficiently compared to the Footprint.

II. LITERATURE SURVEY

1. Overview of Sybil Attacks in VANETs

Douceur, J. R. (2002). "The Sybil Attack". In this seminal work, Douceur discusses the fundamental concept of Sybil attacks in distributed systems, where a single attacker can present multiple identities. The study highlights the challenges of identity verification in peer-to-peer networks, laying the groundwork for understanding such attacks in VANETs.

Ghosh, S., et al. (2016). "A survey of security in VANETs". This comprehensive survey covers various security threats in VANETs, including Sybil attacks. It provides a detailed analysis of the impact of Sybil attacks on VANET communication protocols and the importance of robust detection mechanisms.

2. Detection Techniques Using Proofs of Work

Nakamoto, S. (2008). "Bitcoin: A Peer-to-Peer Electronic Cash System". Nakamoto introduces the concept of Proof of Work (PoW) as a consensus mechanism in blockchain technology. While not specific to VANETs, the principles of PoW can be adapted for verifying the legitimacy of nodes in VANETs, making it harder for attackers to create multiple identities.

Sun, X., & Chen, H. (2014). "A lightweight and efficient re-authentication scheme for VANETs". This paper proposes a PoW-based authentication scheme tailored for VANETs. The authors suggest that PoW can be used to prevent Sybil attacks by requiring computational effort to validate identities, thus deterring attackers from creating numerous fake nodes.

Zhang, X., et al. (2020). "Proof of Work based Sybil attack detection approach in IoT". Although focused on IoT, this study presents a PoW-based approach to detect Sybil attacks, which can be relevant for VANETs. The researchers demonstrate that incorporating PoW can effectively reduce the feasibility of Sybil attacks.

3. Geolocation Techniques for Sybil Attack Detection

Papadimitratos, P., et al. (2008). "Secure vehicular communication systems: Design and architecture". This paper discusses the use of geolocation for securing vehicular communication systems. It highlights the potential of location-based verification methods to identify and mitigate Sybil attacks by cross-referencing reported positions with actual geographic data.

Wang, H., et al. (2013). "Detecting Sybil attacks in VANETs". The authors propose a location-based approach to detect Sybil attacks, using the consistency of location information to verify the legitimacy of nodes. By analyzing the mobility patterns of vehicles, the system can identify anomalies indicative of Sybil attacks.

Lu, R., et al. (2012). "ECPP: Efficient Conditional Privacy Preservation Protocol for VANETs". This study introduces a protocol that combines location verification with privacy preservation. The authors show that geolocation data can be used to detect Sybil attacks while maintaining user privacy, a crucial consideration in VANETs.

III.EXISTING SYSTEM

Zhou et al. [8] proposed a privacy-preserving scheme based on certificates to detect Sybil nodes. The department of motor vehicle (DMV) represents the certificate authority, and is responsible for providing vehicles with a pool of pseudonyms to be used to hide the vehicle's unique identity. The pseudonyms associated with each vehicle are hashed to a common value. An RSU determines whether the pseudonyms come from the same pool by calculating the hashed values of the received pseudonyms. RSUs can detect Sybil nodes and then report such suspected vehicles to DMV.

To resist against RSU compromise, the paper suggests twolevel hash functions with different keys (coarse-grained keys and fine-grained keys). RSU holds each valid coarse-grained key only for a short time which does not know whether the pseudonyms belong to one vehicle or not. If an RSU is compromised, the attacker only gets the coarse-grained hash key for the current time interval while DMV stores all keys and can detect Sybil nodes by two-level hashing. Although deploying trusted certificates is the most efficient approach that can completely eliminate Sybil attacks, it also violates both anonymity and location privacy of entities. Also, relying on a centralized authority to ensure each is assigned exactly one identity which becomes a bottleneck in the large-scale network such as VANETs.

In [30], Chen et al. proposed a group signature-based approach that can be used to enable a member in the group to authenticate himself/ herself anonymously. Meanwhile, if a particular node generates multiple signatures on the same message, the verifier can recognize those signatures. As a result, detecting duplicated signatures signed by the same vehicles can eliminate Sybil attack. However, the malicious vehicle can launch Sybil attack, if he can generate different messages with similar meaning. Recently, Reddy et al. [7] proposed a cryptographic digital signature based method to establish the trust relationship among participating entities.

The most relevant approach to our work is using trajectories of vehicles as its identities to ensure trust between participating nodes. In [32], RSUs broadcasts digital signatures with a timestamp to vehicles which are under its coverage. Vehicles store the RSUs signatures which they gathered in motion. However, since the time stamp is not issued for a dedicated vehicle, a malicious vehicle may claim its presence at certain RSU by merely eavesdropping such broadcasted timestamp on a wireless channel although it may have never been there at that time. In [4], Footprint has been introduced to detect Sybil attack. When a vehicle passes by an RSU, it obtains a signed message as proof of presence at this location at a particular time. A trajectory of a vehicle is a consecutive series of authorized messages collected by the vehicle as it keeps traveling. Sybil attack can be detected using the fact that the trajectories generated by an attacker are very similar. However, Footprint has some critical issues.

Disadvantages

- ❖ The system is not implemented Hashing Keys in order to find Sybil attacks.
- ❖ The system is not implemented attack resistance techniques in order to resist the Sybil and DDOS attacks.

IV.PROPOSED SYSTEM

This paper presents a new method for detecting Sybil attacks by utilizing proofs of work and location. The core concept is that when a vehicle comes across a Roadside Unit (RSU), the RSU should generate an authorized time-stamped tag. This tag is formed by combining the time of the vehicle's arrival and an anonymous location identifier specific to that RSU. The vehicle's trajectory is formed by linking a series of consecutive permitted time-stamped tags in chronological order as it continues to move. The trajectory serves as an anonymous identifier for the vehicle. Given that RSUs are primarily responsible for issuing proof of location to cars, it is crucial for the scheme to be resistant to RSU compromise attacks. To achieve this, we construct the trajectory in such a way that multiple RSUs are capable of creating trajectories for the vehicles, rather than relying on just one RSU. In order to accomplish this, a threshold signature scheme is utilized, ensuring that each Roadside Unit (RSU) can only provide a partial signature for a specific set of time-stamped tags. After a vehicle passes a specific number of RSUs, it can generate a standard signature that proves its position. When the vehicle receives a message that has been authorized by an RSU, it should utilize the message as a starting point to solve a puzzle using a proof-of-work method, which is similar to the one used in Bitcoin [13]. The fundamental concept behind PoW is to furnish RSUs with a verification that confirms the vehicle's accurate resolution of the puzzle. When comparing to Footprint [4], the use of Proof of Work (PoW) restricts the capability of malevolent vehicles to generate different trajectories.

To detect Sybil trajectories, upon receiving an event from other vehicles, the event manager first applies a set of heuristics to construct a connected graph of Sybil nodes, then it uses the maximum clique algorithm [14] to detect all Sybil nodes in that graph.

Advantages

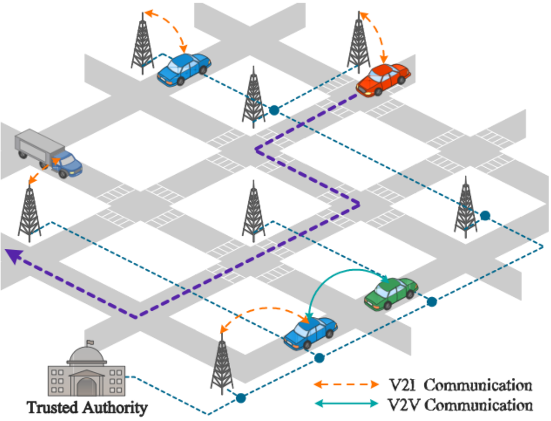
_ The system used threshold signatures to resist RSU compromise attacks. The attacker needs to compromise an infeasible number of RSUs to be able to create fake trajectories.

_ The system used the PoW algorithm with Machine learning classifiers to limit the ability of a malicious vehicle to create multiple forged trajectories, and more importantly, to reduce the detection time for detecting Sybil trajectories which is a critical concern in traffic management applications.

_ The system carefully analyzed the probabilistic nature of PoW based scheme by examining the affecting parameters (e.g travel time between two consecutive RSUs) experimentally, and then we developed a mathematical model that can be used for adjusting these parameters so that the ability of a malicious vehicle to create forged trajectories is reduced significantly.

_ By experiments, we prove that using the proof of work algorithm reduces the ability of a malicious vehicle to maintain actual multiple trajectories simultaneously. Further simulations, analysis, and practical experiments are conducted to evaluate the proposed scheme and compare it with the Footprint [4], the results indicate that the proposed scheme can successfully detect and defend against Sybil attacks in VANETs and more efficiently compared to the Footprint.

V.ARCHITECTURE DIAGRAM:



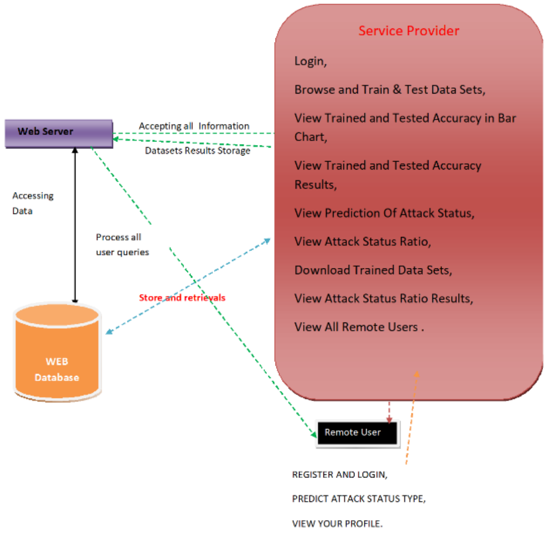

Service Provider
Login,
Browse and Train & Test Data Sets,
View Trained and Tested Accuracy in Bar Chart,
View Trained and Tested Accuracy Results,
View Prediction Of Attack Status,
View Attack Status Ratio,
Download Trained Data Sets,
View Attack Status Ratio Results,
View All Remote Users .
Web Server
Accepting all Information
Datasets Results Storage
Accessing Data
Process all user queries
Store and retrievals
WEB Database
Remote User
REGISTER AND LOGIN,
PREDICT ATTACK STATUS TYPE,
VIEW YOUR PROFILE.

Fig: Architecture diagram

5.1Modules

Service Provider

In this module, the Service Provider has to login by using valid user name and password. After login successful he can do some operations such as

Login, Browse and Train & Test Data Sets, View Trained and Tested Accuracy in Bar Chart, View Trained and Tested Accuracy Results, View Prediction Of Attack Status, View Attack Status Ratio, Download Trained Data Sets, View Attack Status Ratio Results, View All Remote Users.

View and Authorize Users

In this module, the admin can view the list of users who all registered. In this, the admin can view the user’s details such as, user name, email, address and admin authorizes the users.

Remote User

In this module, there are n numbers of users are present. User should register before doing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user will do some operations like REGISTER AND LOGIN, PREDICT ATTACK STATUS TYPE, VIEW YOUR PROFILE.

VI.ALGORITHMS:

Decision tree classifiers

Decision tree classifiers are used successfully in many diverse areas. Their most important feature is the capability of capturing descriptive decision making knowledge from the supplied data. Decision tree can be generated from training sets. The procedure for such generation based on the set of objects (S), each belonging to one of the classes C1, C2, …, Ck is as follows:

Step 1. If all the objects in S belong to the same class, for example Ci, the decision tree for S consists of a leaf labeled with this class
Step 2. Otherwise, let T be some test with possible outcomes O1, O2,…, On. Each object in S has one outcome for T so the test partitions S into subsets S1, S2,… Sn where each object in Si has outcome Oi for T. T becomes the root of the decision tree and for each outcome Oi we build a subsidiary decision tree by invoking the same procedure recursively on the set Si.

Gradient boosting

Gradient boosting is a machine learning technique used in regression and classification tasks, among others. It gives a prediction model in the form of an ensemble of weak prediction models, which are typically decision trees.[1][2] When a decision tree is the weak learner, the resulting algorithm is called gradient-boosted trees; it usually outperforms random forest.A gradient-boosted trees model is built in a stage-wise fashion as in other boosting methods, but it generalizes the other methods by allowing optimization of an arbitrary differentiable loss function.

K-Nearest Neighbors (KNN)

- Simple, but a very powerful classification algorithm
- Classifies based on a similarity measure
- Non-parametric
- Lazy learning
- Does not “learn” until the test example is given

- Whenever we have a new data to classify, we find its K-nearest neighbors from the training data

Logistic regression Classifiers

Logistic regression analysis studies the association between a categorical dependent variable and a set of independent (explanatory) variables. The name *logistic regression* is used when the dependent variable has only two values, such as 0 and 1 or Yes and No. The name *multinomial logistic regression* is usually reserved for the case when the dependent variable has three or more unique values, such as Married, Single, Divorced, or Widowed. Although the type of data used for the dependent variable is different from that of multiple regression, the practical use of the procedure is similar.

Logistic regression competes with discriminant analysis as a method for analyzing categorical-response variables. Many statisticians feel that logistic regression is more versatile and better suited for modeling most situations than is discriminant analysis. This is because logistic regression does not assume that the independent variables are normally distributed, as discriminant analysis does.

This program computes binary logistic regression and multinomial logistic regression on both numeric and categorical independent variables. It reports on the regression equation as well as the goodness of fit, odds ratios, confidence limits, likelihood, and deviance. It performs a comprehensive residual analysis including diagnostic residual reports and plots. It can perform an independent variable subset selection search, looking for the best regression model with the fewest independent variables. It provides confidence intervals on predicted values and provides ROC curves to help determine the best cutoff point for classification. It allows you to validate your results by automatically classifying rows that are not used during the analysis.

Naïve Bayes

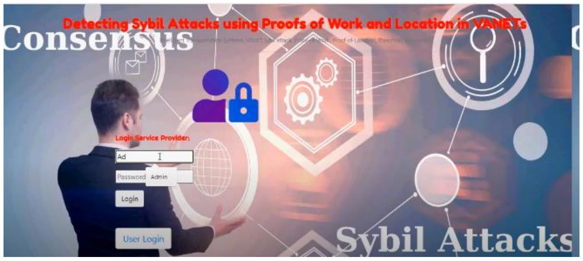
The naive bayes approach is a supervised learning method which is based on a simplistic hypothesis: it assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature .

Yet, despite this, it appears robust and efficient. Its performance is comparable to other supervised learning techniques. Various reasons have been advanced in the literature. In this tutorial, we highlight an explanation based on the representation bias. The naive bayes classifier is a linear classifier, as well as linear discriminant analysis, logistic regression or linear SVM (support vector machine). The difference lies on the method of estimating the parameters of the classifier (the learning bias).

While the Naive Bayes classifier is widely used in the research world, it is not widespread among practitioners which want to obtain usable results. On the one hand, the researchers found especially it is very easy to program and implement it, its parameters are easy to estimate, learning is very fast even on very large databases, its accuracy is reasonably good in comparison to the other approaches. On the other hand, the final users do not obtain a model easy to interpret and deploy, they does not understand the interest of such a technique.

Thus, we introduce in a new presentation of the results of the learning process. The classifier is easier to understand, and its deployment is also made easier. In the first part of this tutorial, we present some theoretical aspects of the naive bayes classifier. Then, we implement the approach on a dataset with Tanagra. We compare the obtained results (the parameters of the model) to those obtained with other linear approaches such as the logistic regression, the linear discriminant analysis and the linear SVM. We note that the results are highly consistent. This largely explains the good performance of the method in comparison to others. In the second part, we use various tools on the same dataset (**Weka 3.6.0**, **R 2.9.2**, **Knime 2.1.1**, **Orange 2.0b** and **RapidMiner 4.6.0)**. We try above all to understand the obtained results.

VII. SCREENSHOTS:

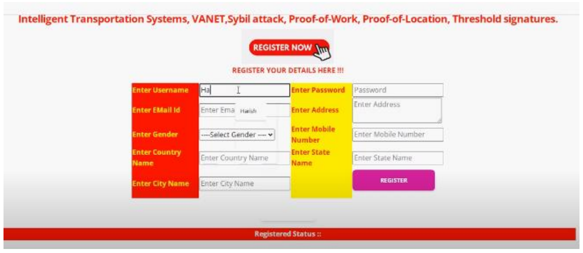


Detecting Sybil Attacks using Proofs of Work and Location in VANETs

Login Service Provider:

Ad

Password Admin

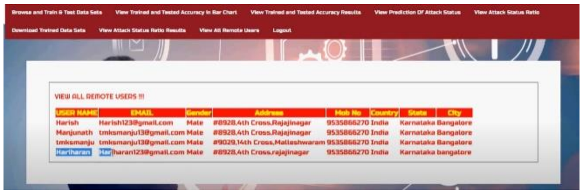


Intelligent Transportation Systems, VANET,Sybil attack, Proof-of-Work, Proof-of-Location, Threshold signatures.

REGISTER YOUR DETAILS HERE !!!

Enter Username	Ha	Enter Password	Password
Enter EMail Id	Enter Ema Harish	Enter Address	Enter Address
Enter Gender	----Select Gender ----	Enter Mobile Number	Enter Mobile Number
Enter Country Name	Enter Country Name	Enter State Name	Enter State Name
Enter City Name	Enter City Name		REGISTER

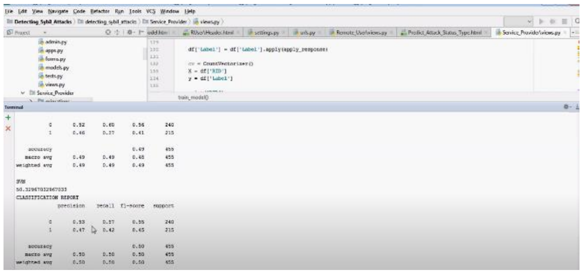
Registered Status ::



Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Attack Status
View Attack Status Ratio
Download Trained Data Sets
View Attack Status Ratio Results
View All Remote Users
Logout

VIEW ALL REMOTE USERS !!!

USER NAME	EMAIL	Gender	Address	Mob No	Country	State	City
Harish	Harish123@gmail.com	Male	#8928,4th Cross,Rajajinagar	9535866270	India	Karnataka	Bangalore
Manjunath	tmksmanju13@gmail.com	Male	#8928,4th Cross,Rajajinagar	9535866270	India	Karnataka	Bangalore
tmksmanju	tmksmanju13@gmail.com	Male	#9029,14th Cross,Malleshwaram	9535866270	India	Karnataka	Bangalore
Hariharan	Hariharan123@gmail.com	Male	#8928,4th Cross,rajajinagar	9535866270	India	Karnataka	bangalore

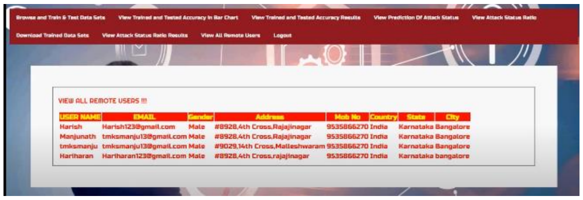


File Edit View Navigate Code Refactor Run Tools VCS Window Help
Detecting_Sybil_Attacks › detecting_sybil_attacks › Service_Provider › views.py
Project
add.html
RUser\Header.html
settings.py
urls.py
Remote_User\views.py
Predict_Attack_Status_Type.html
Service_Provider\views.py
admin.py
apps.py
forms.py
models.py
tests.py
views.py
Service_Provider
Terminal

```
df['Label'] = df['Label'].apply(apply_response)

cv = CountVectorizer()
X = df['RID']
y = df['Label']
```

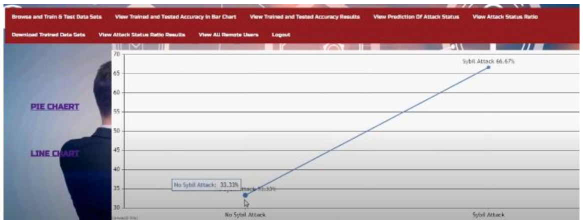
train_model()

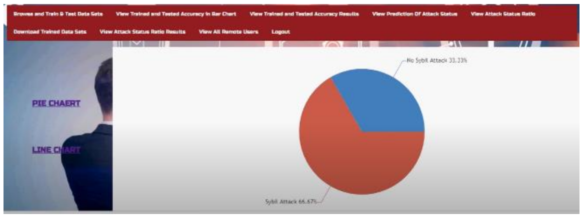
```
           0       0.52      0.60      0.56       240
           1       0.46      0.37      0.41       215

    accuracy                           0.49       455
   macro avg       0.49      0.49      0.48       455
weighted avg       0.49      0.49      0.49       455

SVM
50.32967032967033
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.53      0.57      0.55       240
           1       0.47      0.42      0.45       215

    accuracy                           0.50       455
   macro avg       0.50      0.50      0.50       455
weighted avg       0.50      0.50      0.50       455
```


Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Attack Status
View Attack Status Ratio
Download Trained Data Sets
View Attack Status Ratio Results
View All Remote Users
Logout

VIEW ALL REMOTE USERS !!!

USER NAME	EMAIL	Gender	Address	Mob No	Country	State	City
Harish	Harish123@gmail.com	Male	#8928,4th Cross,Rajajinagar	9535866270	India	Karnataka	Bangalore
Manjunath	tmksmanju13@gmail.com	Male	#8928,4th Cross,Rajajinagar	9535866270	India	Karnataka	Bangalore
tmksmanju	tmksmanju13@gmail.com	Male	#9029,14th Cross,Malleshwaram	9535866270	India	Karnataka	Bangalore
Hariharan	Hariharan123@gmail.com	Male	#8928,4th Cross,rajajinagar	9535866270	India	Karnataka	bangalore


Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Attack Status
View Attack Status Ratio
Download Trained Data Sets
View Attack Status Ratio Results
View All Remote Users
Logout
PIE CHAERT
LINE CHART
70
65
60
55
50
45
40
35
30
Sybil Attack 66.67%
No Sybil Attack: 33.33%
No Sybil Attack
Sybil Attack


Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Attack Status
View Attack Status Ratio
Download Trained Data Sets
View Attack Status Ratio Results
View All Remote Users
Logout
PIE CHAERT
LINE CHART
No Sybil Attack 33.33%
Sybil Attack 66.67%

VIII.CONCLUSION

Sybil attacks can have catastrophic implications in Vehicular Ad Hoc Networks (VANETs). This paper presents a new method for identifying Sybil attacks by utilizing proofs of work and location. Sybil attacks in VANETS can have catastrophic implications since they can create an anonymous trajectory of a vehicle by gathering a series of location proofs from several RSUs. This paper presents a new method for identifying Sybil attacks by utilizing proofs of work and location. A vehicle's anonymous trajectory is created by collecting a sequential record of locations from several Roadside Units (RSUs) that it comes across. To mitigate the RSU compromise attack, it is necessary to have a minimum of t RSUs instead of just one to issue allowed messages for vehicles and create a proof of location message using threshold signature. Additionally, the implementation of a proof-of-work method can restrict the capacity of malevolent cars to generate counterfeit trajectories. Our assessments have shown that our technique is highly effective in detecting Sybil attacks, with a high detection rate and a low percentage of false negatives. Furthermore, the communication and computation overhead of the exchanged packets is deemed acceptable.

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, “Privacy-preserving 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.

[11] M. S. Bouassida, G. Guette, M. Shawky, and B. Ducourthial, “Sybil nodes detection based on received signal strength variations within vanet.” IJ Network Security, vol. 9, no. 1, pp. 22–33, 2009.

[12] S. Syed and M. E. Cannon, “Fuzzy logic-based map matching algorithm for vehicle navigation system in urban canyons,” in ION National Technical Meeting, San Diego, CA, vol. 1, 2004, pp. 26–28.

[13] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008.

[14] E. Tomita, Y. Sutani, T. Higashi, S. Takahashi, and M. Wakatsuki, “A simple and faster branch-and-bound algorithm for finding a maximum clique,” in International Workshop on Algorithms and Computation. Springer, 2010, pp. 191–203.

[15] M. Alsabaan, W. Alasmary, A. Albasir, and K. Naik, “Vehicular networks for a greener environment: A survey.” IEEE Communications Surveys and Tutorials, vol. 15, no. 3, pp. 1372–1388, 2013.

[16] A. Shamir, “How to share a secret,” Communications of the ACM,vol. 22, no. 11, pp. 612–613, 1979.