ISSN (Online) : 2319 - 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

Volume 3, Special Issue 3, March 2014

2014 International Conference on Innovations in Engineering and Technology (ICIET'14)
On 21st & 22nd March Organized by

K.L.N. College of Engineering, Madurai, Tamil Nadu, India

A Secure Communication for Clustered RSU in VANETs

G.J.Archanaa[1], R.Venittaraj[2]

[1]PG Scholar , Department of Computer and Communication, Mepco Schlenk Engineering College, Tamilnadu, India

[2] Senior Assistant Professor, Department of Information Technology, Mepco Schlenk Engineering College, Tamilnadu, India

Abstract—Vehicular ad hoc networks (VANETs) enable vehicles to communicate with each other but require efficient and robust routing protocols for their success. In this paper, we exploit the infrastructure of roadside units (RSUs) to efficiently and reliably route packets in VANETs. Our system operates by using vehicles to carry and forward messages from a source vehicle to a nearby RSU and, "also adding a digital signature by advanced encryption standards to every node for the security purpose". RSU's are formed as cluster for consuming less energy and life time also increases. We evaluate the performance of our system using the ns2 simulation platform and compare our scheme to existing solutions. Then traffic is generated by VANETMOBISIM software. The results prove the feasibility and efficiency of our scheme**.**

Keywords—Carry and forward, geographic forwarding, intervehicle communication, roadside units (RSUs), routing, vehicular ad hoc networks (VANETs), Digital Signature.

I. INTRODUCTION

The emergence of mobile ad hoc networks, researchers have conceptualized the idea of communing vehicles, giving rise to vehicular ad hoc networks (VANETs), which are the main focus of engineers yearning to turn cars into intelligent machines that commune for safety and comfort purposes. A VANET is formed by vehicles that are equipped with wireless communication devices, positioning systems, and digital maps. VANETs also allow vehicles to connect to roadside units (RSUs), which are connected to the Internet and may also be connected with each other via a high-capacity mesh network. Several researchers have developed unicast routing protocols for VANETs, some of which use a position-based greedy approach that uses the geographic coordinates of vehicles to find a good route. Another set of protocols, which use carry-and-forward approaches and aim to route packets in sparse VANETs, are called delay-tolerant algorithms [1].

The basic motivation behind using RSUs to route packets is that RSUs are a fixed infrastructure. It is much easier to send a packet to a fixed target than to a remote moving object. In addition, the delay of sending the packet through the fixed RSU network is much less than through the VANET. We call our approach Carry and forwArd mechaNisms for Dependable mEssage deLIvery in VanEts using Rsus (CAN DELIVER). The design of our system is divided into two basic parts: the first part governs routing from a vehicle to its nearest RSU, and the second part handles routing from RSUs to vehicles.

II. PROPOSED FRAMEWORK

A. Motivations and Assumptions

When exploring the problem of routing in VANETs, it is very important to distinguish between two different cases. The first occurs when the destination is hundreds of meters far from the source. Hence, the sender and the receiver are a few hops apart. The second case occurs when the destination is very far from the source (a few to tens of kilometers). The first case has been studied in detail, and a variety of protocols (see Section V) that route packets to near and medium locations have been proposed. Our objective is to efficiently route packets to distant locations. When a vehicle S needs to send a packet P to a far away vehicle D, several problems need to be solved:

1. Since D is moving, it is hard to specify its location at the time it receives P. Existing approaches use

geocasting to send P to the area *A* in which D is moving, where it will be broadcasted to all vehicles. The problem is that as the distance from S to D

2. increases, *A* rapidly grows. Our solution reduces the size of *A* as much as possible.
3. In sparse settings, P may be carried for long times by intermediate vehicles. To solve this, a vehicle in our system tries to forward P to more than one neighbor (possibly in different directions).
4. In many cases, P might be carried by a vehicle that moves away from D without meeting any vehicles to forward P to. This might lead to delaying P or to dropping it. To solve this, we use controlled redundancy to increase the chances of reaching D without causing significant overhead.

CAN DELIVER depends on the system of RSUs to relay packets to distant locations. Several approaches have been proposed for designing the RSU network. In all cases, each RSU has a way to connect (directly or via other RSUs) to any other RSU.

The Fig.1 shows that a Road Side Unit (RSU) can communicate with other nearby RSU and neighborhood vehicles can also communicate with RSU. The moving vehicles location get from VANETMOBISIM software can get the information about vehicles location, speed, direction of each and every vehicles in the road way. we can communicate vehicle to vehicle and vehicle to RSU is done through CANDELIVER protocol and follow the mechanism in these following algorithms.

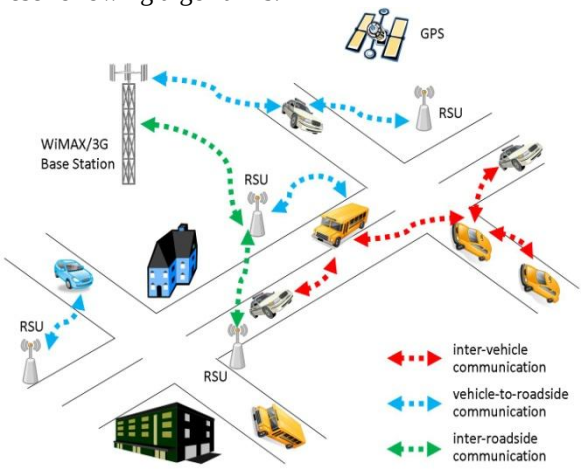

Fig 1.Simple Representation of RSU in VANET

B. Global Overview

To preserve their privacy, vehicles could employ pseudonyms instead of actual IDs. In addition, each vehicle periodically changes its pseudonym. In CAN DELIVER, we employ a strategy similar to [2] for vehicles to create and change their pseudonyms. Each vehicle periodically sends Hello packets to its neighbors, and maintains a list *L* of pseudonyms, positions, speeds, directions, and timestamps of vehicles in its vicinity. After this, each vehicle includes in a Hello its LSDT as well as the data it has in *L*. Each vehicle that receives a Hello will add an entry in the Hello to L. The Hello Packets are exchanged between vehicles and also send to RSU,has send beacon packets Although in CAN DELIVER many control packets (Hello and beacons) are exchanged, they do not produce large. overhead since they are small in size and no fixed routing paths are cached. If the distance between itself and the vehicle of the entry is less than a threshold d_{th}. Hence, S will send the packet to neighbors that are nearest to the "Destination Waypoint." For example, in Fig. 3, vehicle V_1 wants to send a packet to RSU R. It obtains from its the shortest path from its location to R, which consists of the waypoints *{*W_S, A, B, C, E, F, G, and W_D*}*, and calculates the distance from its current position to R as the sum of individual distances between consecutive waypoints that constitute the shortest path plus (or minus) the distance between its current location and the first waypoint. In Fig. 2, the distance between V_1 and R will be calculated by V_1 as {AB+WA}

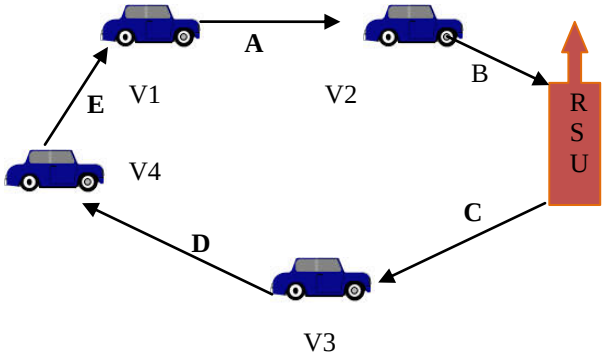

Fig 2. Sending a packet from a vehicle to an RSU

III.ANALYSIS

Resending a Packet: If S sends a packet P to R and does not receive an ACK within a time T_w , it resends P. T_w can be estimated as follows: if v is the average speed in the area S is moving in, r is the transmission range, d is the length of the path from S to R, c is the transmission time between two neighbors, d^* is the average distance during which a vehicle carries P, and t_s is a safety time factor, then T_w is set to be equation given below,

$$\mathbf{Tw=2 * [(d/(r+d^*)) * (c+(d^*/v)+ ts]} \qquad (1)$$

which is the time spent while a vehicle is carrying P (d^*/v) plus the time in which P is being sent to a neighbor, multiplied by the number of hops to reach R, with a safety time factor added. This total value is multiplied by 2 since the ACK needs around the same time to reach S. In CAN DELIVER, RSUs are assigned the task of calculating and distributing the value of d^* that will be used by vehicles to estimate T_w value of d in the next beacon.

A. Defining a path from a vehicle to an RSU Packet P(input)

Vehicle S(Source)
RSU R(Destination)
Start:

Step1:
if(R is within S Range)
S sends directly to R
Go to end
else
Step2:S defines total path between itself and R
Step3:S determines the set Sn of neighbors that are nearer from it to R
Step4: if (Sn=Ǿ)
Go to Delay Routine
else
S sends P to k neighbors in Sn
S drops P
End.

Fig.3.Example of an RSU choosing a road to send a packet to

The following are the algorithm for vehicles are sending a packet to nearby RSU which is done by this algorithm

B. Algorithm for sending a packet from an RSU to a vehicle. Part 1: at the RSU.

Packet P (input)
Vehicle S(Source)
RSU R (Destination)
Start:
Step1: R calculates the estimated distance travelled by D from the instance it sent its last beacon until it receives P
Step 2: R determines estimated location of D at the time it receives P
step 2.1: R determines whether the location should be calculated using CASE1 or CASE2
step 2.2: R calculates the center and radius of the estimated area(Ag)
step 2.3: R adds the center and radius of Ag to P
Step 3: R sends to K vehicles
step 3.1: R determines the best candidate roads
step 3.2: R forwards P to the farthest K vehicles moving on the candidate roads
End

Table1: Variable and Acronyms used in this paper

Variable	Definition
T	Simulation Time
S	Source Vehicle
D	Destination Vehicle
RSU	Road Side Unit
N	Neighbor Vehicle
r	Transmission Range
d	Distance from RSU to Vehicle
d*	Average vehicle during which a vehicle will carry a packets
c	Number of vehicles per community
v	Vehicle average speed
ts	Maximum time
Ǿ	Empty
Sn	Set of neighbors
P	Packet
tn	Timestamp
d_p	Coordinates of the estimated area
dtot	Total distance travelled by vehicle between timestamp
V	Vehicle
d_{th}	Threshold

C. Operations of Vehicles Estimated Area:

When a vehicle V receives a packet P destined to a vehicle D, it checks if it has in *L* an entry for D with a timestamp t_n greater than t_1 (i.e., fresher information about D's location). If this is the case, V calculates D's A_E and updates P with the new values of and in addition to t_n. V moves P to a queue and frequently checks for new neighbors. This operation continues until a vehicle that is within A_E receives P.The operations made by vehicles outside A_E. When a vehicle (V) within A_E receives P, it checks if D is a neighbor, and sends P to D, if this is the case. Else, it broadcasts P to all its neighbors.

D.RSU to Vehicle Algorthim:Part2

Packet P(input)
RSU R (Source)
vehicle D(Destination)
vehicle N(Intermediate vehicle carrying P)
Start:
Step1: if(D is a neighbor of N)
N sends P directly to D

Go to end

Step 2: N updates the estimated area(Ag)
Step 3: if (N has one or more neighbor within Ag)
N sends P to these neighbors.
N drops P

Step 4: if (D is neighbor of N)
N send P directly to D
Go to end
else
Step 4.1: if(N has one or more neighbors)
Step 4.2: N broadcast P to all its neighbors
else
N stores P to queue

End

These are the algorithm for RSU to vehicle in the moving vehicles and we have to calculate distance from RSU to vehicles in the traffic methodology. For calculating the distance from vehicles to nearby RSU means using the formula we calculate the distance

$$\mathbf{d_t = (v1 * (tc - t1)) + d_s} \quad (2)$$

where, v1 is speed, tc is Current time,t1 is Timestamp, d_s is error Factor. The difference between timestamp and the current time as multiplied by the speed of moving vehicles and add some of error factor to it then we got the distance between the RSU and the vehicles. Distance Travelled by vehicle between tc and tf instances is

$$\mathbf{dc = v1 * tf} \quad (3)$$

When a vehicle within AE receives P, it checks if D is a neighbor, and sends P to D, if this is the case. Else, it broadcasts P to all its neighbors. If a vehicle outside AE receives P, or if a vehicle receives P for a second time, it drops it. For calculating Radius,

$$\mathbf{r = (0.5 * d_{tot})} \quad (4)$$

Where, **dtot = dt + dc**. A vehicle that receives P and is not within AE can distinguish whether it should carry and forward P to AE or drop it by checking if the neighbor that sent P to it is within AE. A vehicle within AE that forwards P to its neighbors will drop P only if at least one neighbor.

IV. ADDING A DIGITAL SIGNATURE

A digital signature is an electronic form of a signature that can be used to authenticate the identity of the sender of a message or the signer of a document, and also ensure that the original content of the message or document that has been sent is unchanged.

Digital signatures are easily transportable and cannot be imitated by someone else. In the VANET, the nodes are transferring the packets while sending a packet it need security so that we plan to add Digital signature in route discovery mechanism. That means instead of sending node (vehicle) id every node create digital signature based on its node-id also we plan to add hashing mechanism with signature to improve the security level. For improving the security in the traffic message adding the digital signature to the message by adding the signature user's message or data are delivered in secure manner by using Mixing columns in Advanced Encryption standars.

1. A digital signature (not to be confused with a digital certificate) is a signature that can be used to authenticate the identity of the sender of a message, and possibly to ensure that the original content of the message or document that has been sent is unchanged.
2. A digital signature can be used with any kind of message, whether it is encrypted or not, simply so that the receiver can be sure of the sender's identity and that the message arrived intact.

A. Advanced encryption standards

AES is a symmetric block cipher with a block size of 128 bits. Key lengths can be 128 bits, 192 bits, or 256 bits; 8 called AES-128, AES-192, and AES-256, respectively. AES-128 uses 10 rounds, AES-192 uses 12 rounds, and AES-256 uses 14 rounds.

The main loop of AES performs the following functions:

a. Sub Bytes()
b. Shift Rows()
c. Mix Columns()
d. Add RoundKey()

The first three functions of an AES round are designed to thwart cryptanalysis via the methods of "confusion" and "diffusion." The fourth function actually encrypts the data. Diffusion means patterns in the plaintext are dispersed in the cipher text. Confusion means the relationship between the plaintext and the cipher text is obscured.

A simpler way to view the AES function order is:

1. Scramble each byte (SubBytes).
2. Scramble each row (ShiftRows).
3. Scramble each column (Mix Columns).
4. Encrypt (Add RoundKey).

A term associated with AES is "the State," an 'intermediate cipher,'11 or the cipher text before the final round has been applied. AES formats plaintext into 16 byte (128-bit) blocks, and treats each block as a 4x4 State array. It then performs four operations in each round. The arrays contains row and column information used in the operations, especially Mix Columns() and Shift rows().

B. Mix Columns

Mix Columns() also provides diffusion by mixing data within columns. The 4 bytes of each column in the State are treated as a 4-byte number and transformed to another 4-byte number via finite field mathematics.

Generate the random key and values are mixing with each other that value is added to the message and exchanged to the nearby vehicles in the network.

V.CLUSTERING METHODOLOGY

Clustering is an important mechanism in large multi-hop wireless sensor networks for obtaining scalability, reducing energy consumption and achieving better network performance. Here increases the number of RSU we get more delivery rate and reduces traffic jam also.

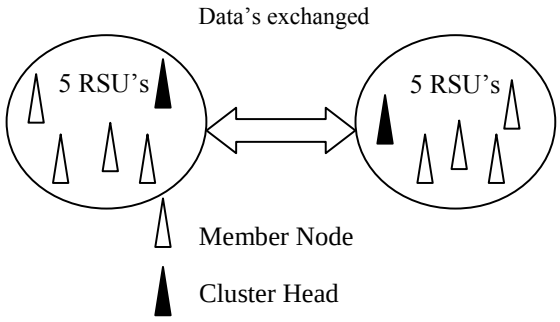

Fig 4 Formation of cluster nodes

so we are cluster the RSU for energy consumption first time of transmission in the cluster group any one of the node act as a cluster head and the next time of transmission the cluster head will be changed each and every time cluster head will be selected based on energy level of node. By these methods we can improve the energy level of RSU for long time. For improve the efficiency and less consumption of energy using these cluster techniques.

The cluster formation method is very simple. The random node is elected as the cluster head. The nodes use the information in the hello packets to decide whether or not they are the cluster heads. The cluster head regards all nodes it has bi-directional links to as its member nodes. Clusters are identified by their respective cluster heads, which means that cluster head must changes as infrequently as possible. A node regards itself as a member node to a particular cluster if it has a bi-directional link to the cluster head. It is possible for a node to belong to several clusters.

For creating the number of nodes and generating the traffic in the network by using the VANET MOBISIM and send the hello packets to each and every nodes in the network and stored in the table. Although in CANDELIVER many control packets such as HELLO and Beacon are exchanged, they do not produce large overhead since they are small in size and no fixed routing paths are cached. Both analysis and simulations prove that our system produces small overall traffic. Data's are forwarded to RSU-to-Vehicles and further it transfer the packets for the security purpose we are adding the digital signature using the AES in the VANETs and also it applied to the clustering methods in CANDELIVER protocol.

IV.PERFORMANCE ANALYSIS

This section presents the simulations that were performed to evaluate CAN DELIVER using the network simulator ns2 software and VANETMOBISIM for generate the traffic path and node mobility in the network. The proposed method has been implemented and tested on various methods. The visual results of sample VANETMOBISIM generated the traffic path for sensor nodes and their position. Nam output generated after the CANDELIVER protocol is implemented the sensor nodes are deployed randomly help of VANETMOBISIM software. And it shows the RSU nodes in between sensor nodes in VANET. It shows the evaluates the performance of the proposed system by graphical measures

A. Varying the Number of RSUS and Simulating Traffic Jams

An important parameter that highly affects the performance of CAN DELIVER is the number of RSUs (N_{RSU}). In this section, we simulated scenarios with 1, 5, 10, and 20 RSUs uniformly distributed across the network.

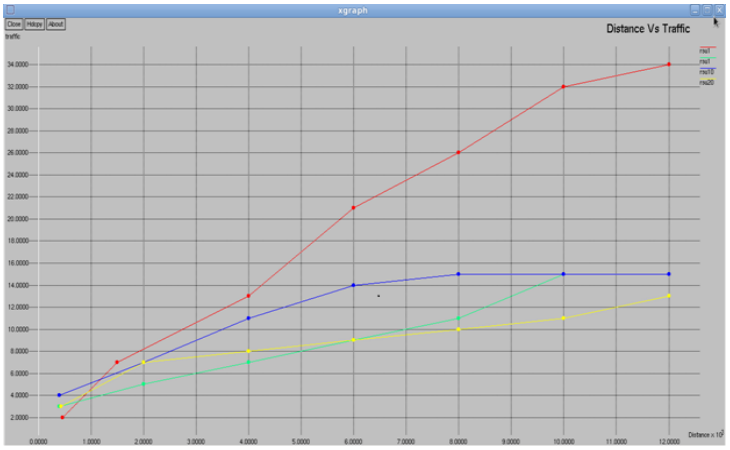

Fig 5. Reduces Traffic jam by increases the No of RSU

By increases the number of RSU we can increases the delivery rate i.e., Number of RSU id Directly proportional to the traffic jam more RSU are deployed means traffic jam will reduced and most important is deployment of RSU is based on the road condition. For example In metro cities we deploy in traffic signal light it enough for communication but in Highway we deploy RSU in particular distance to distance in roadway.

V. CONCLUSION AND FUTURE WORK

This paper has presented CAN DELIVER, which is part of a complete system that we are designing for providing car drivers and passengers pervasive access to needed data while on the road. The evaluation of CAN DELIVER confirmed its effectiveness as compared to recent routing protocols for VANETs. Ongoing work is focusing on devising secure mechanisms for registering users to the system of RSUs and designating them as

proxies to Internet service providers that provide data to these users. A preliminary design and implementation of such mechanisms were published recently in [9].

Planned future work relates to designing bundling methods for allowing RSUs to deliver the maximum amount of possibly heterogeneous data to users. we plan to add Digital signature in route discovery mechanism. That means instead of sending node (vehicle) id every node create digital signature based on its node-id also we plan to add hashing mechanism with signature to improve the security level. Security for the Cluster formation of RSU nodes which can be deployed in signal light or the depends upon the road condition so improve the reliability and security adding some more mechanism .In future work to send the data in secure and reliable manner.

REFERENCES

[1] "A static-node assisted adaptive routing protocol in vehicular networks," in *Proc. VANET*, New York, Sep. 2007,

[2] "TrafRoute: A different approach to routing in vehicular networks," in *Proc. VECON*, Niagara Falls, ON, Canada, 2010.

[3] "VANET routing on city roads using real-time vehicular traffic information,*" IEEE Trans. Veh. Technol.*

[4] Q. Song and X. Wang, "Efficient routing on large road networks using hierarchical communities," *IEEE Trans. Intell. Transp. Syst.*, vol. 12,no. 1, pp. 132–140, Mar. 2011.

[5] Skordylis and N. Trigoni, "Efficient data propagation in trafficmonitoring vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 12,no. 3, pp. 680–694, Sep. 2011.

[6] Y. Bae and N. H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," in *Proc. MobiCom*, Dallas, TX, Oct. 1998, pp. 66–75.

[7] S. Basagni, I. Chlamtac, and V. R. Syrotiuk, "A distance routing effect algorithm for mobility (DREAM)," in *Proc. MobiCom*, Dallas, TX, Oct. 1998, pp. 76–84.

[8] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "MDDV: Mobilitycentric data dissemination algorithm for vehicular networks," in *Proc. VANET*, Philadelphia, PA, 2004, pp. 47–56.

[9] Casteigts, A. Nayak, and I. Stojmenovic, "Communication protocolsfor vehicular ad hoc networks," *Wirel. Commun. Mobile Comput.*, vol. 11,no. 5, pp. 567–582, May 2011.

[10] K. Mershad and H. Artail, "REACT: Secure and efficient data acquisition in VANETs," in *Proc. WiMob*, Shanghai, China, Oct. 2011, pp. 149–156.

[11] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks", Computer Communications, vol. 30, no. 14-15, pp. 2826-2841, 2007.

[12] Zhenya Zhang,"Clustering aggregation based on genetic algorithm for documents clustering",vol no.13, pp 598-610,june 2011.

[13] R. Ahlswede, N. Cai, S. Li, and R. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[14] S. Y. Oh, M. Gerla, and A. Tiwari, "Robust MANET routing using adaptive path redundancy and coding," in *Proc. COMSNETS*, Bangalore, India, 2009, pp. 224–233.

[15] Z. Li, Y. Zhu, and M. Li, "Practical location-based routing in vehicular adhoc networks," in *Proc. MASS*, Macau, China, Oct. 2009, pp. 900–905.

[16] I. Leontiadis, G. Marfia, D. Mack, G. Pau, C. Mascolo, and M. Gerla, "On the effectiveness of an opportunistic traffic management system for vehicular etworks," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1537–1548, Dec. 2011.