Impact Factor (SJIF): 4.542

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 4, Issue 7, June-2017

SOFTWARE DEFINED NETWORKING WITH PSEUDONYM SYSTEMS FOR SECURE VEHICULAR CLOUDS

Darshitha D R[1], Veerappa B N[2]

[1] *Department of Studies in Computer Science Engineering, UBDTCE, Davangere.*

Abstract — The vehicular cloud is a promising new paradigm where vehicular networking and mobile cloud computing are elaborately integrated to enhance the quality of vehicular information services. In this paper, we exploit software-defined networking technology to significantly extend the flexibility and programmability for pseudonym management in vehicular clouds. And a software-defined pseudonym system where the distributed pseudonym pools are promptly scheduled and elastically managed in a hierarchical manner. In order to decrease the system overhead due to the cost of inter-pool communications, we leverage the two-sided matching theory to formulate and solve the pseudonym resource scheduling.

I. INTRODUCTION

With the rapid development of wireless communication technologies vehicles can utilize vehicle-to-infrastructure and vehicle-to-vehicle communications with the help of on-board devices to form vehicular networks. However, many emerging mobile applications require larger and secure storage and complex computation, and bring new resource challenges to vehicular networks, e.g., vehicle platoon, real-time video streaming application and vehicular augmented reality, social media sharing. To meet the growing demands of radio and computing resources, vehicular networks take the advantages of cloud computing and are evolving towards vehicular clouds. From a system-level view, idle resources in vehicles, network infrastructures (e.g., road-side unit (RSU)) and cloud infrastructures (e.g., data centre) can be recruited to form a vehicular cloud system. A typical vehicular cloud system consists of three different levels as following.

1) At the bottom level, cooperative vehicles create a vehicular cloud.

2) At the middle layer, a set of adjacent RSUs form a local cloud.

3) At the top layer, central cloud manages resources in the system.

While ubiquitous wireless communication of pervasive cloud computing greatly facilitate the formation and functioning of vehicular cloud, privacy and security challenges remain to be addressed for this new domain To secure vehicular clouds, we focus on pseudonym, which is an essential resource for vehicles to protect location privacy. Most of the privacy protection schemes are implemented on the basis of pseudonyms, e.g., group signature, silent period, and mix-zone. Vehicles should periodically change their pseudonyms to avoid being continuously tracked. Moreover, a third-party cloud service provider may pose potential threats to the vehicles because of data leakage. This further highlights the importance of pseudonyms for vehicles to protect privacy in vehicular clouds. Vehicles need to possess sufficient pseudonyms to be able to frequently change for anonymity.

II. LITERATURE SURVEY

A. Introduction to Cooperative download in Vehicular Environments

Vehicular networks are expected to deploy short-range communication technology for inter-vehicle communications. In addition to vehicle-vehicle communication, users will be interested in accessing the multimedia-rich Internet from within the vehicular network. Conventional client-server approaches in the face of intermittent connectivity would experience degraded performance. A new paradigm in content delivery on the Internet using peer-peer swarming protocols is emerging. The goal of the Internet swarming protocols is to reduce the load on content servers.

B. Applications of VANETs

In vehicular ad hoc networks (VANETs), a broad range of applications can be classified as data dissemination services, such as traffic information broadcasting, entertainment content downloading, and commercial advertising. Due to the high cost and low speed of data dissemination through wireless wide area networks (WWANs) such as cellular

networks, data dissemination based on roadside units (RSUs) is widely investigated in the existing research following the concept of drive-thru Internet.

In order to improve the efficiency of data dissemination, increasing the availability of the RSUs is indispensable. Based on the concept of wireless metropolitan area sharing networks (WMSNs), the APs of the publicly and/or privately owned roadside WLANs (RS-WLANs) can be shared and functions as RSUs. Because of the bandwidth limitation of the wireless connections from the Internet to the RS-WLANs (e.g., in the residential area), a message pre-downloading mechanism can be implemented where messages are pre-downloaded to the RSUs and scheduled for transmission upon the visit of a vehicle.

Consider a VANET with sparsely located RS-WLANs and investigate the optimal scheduling problem when a vehicle travels through a specific trajectory consists of a set of RS-WLANs pre-downloaded with network coded messages. The objective is to maximize the delivery probability while minimize the delivery delay of the data dissemination sessions. The optimal scheduling problem is mathematically formulated by considering the vehicle trajectory, resource allocation model, and message pre-downloading profile.

C. Content Downloading in VANETs

Content downloading in vehicular networks is a topic of increasing interest: services based upon it are expected to be hugely popular and investments are planned for wireless roadside infrastructure to support it. Content downloading system leveraging both infrastructure to vehicle and vehicle to vehicle communication. With the goal to maximize the system throughput, we formulate a max flow problem that ac-counts for several practical aspects, including channel contention and the data transfer paradigm. Through this study, the factors that have the largest impact on the performance and derive guidelines for the design of the vehicular network and of the roadside infrastructure supporting it.

Previous works on content downloading in vehicular net-works have dealt with individual aspects of the process, such as the deployment of roadside APs, the performance evaluation of I2V communication, the network connectivity, or the exploitation of specific V2V transfer paradigms. None of them, however, has tackled the problem as a whole, trying to quantify the actual potential of an I2V/V2V-based content downloading. In order to fill such a gap, the following questions are posed: (i) which is the maximum downloading performance theoretically achievable through DSRC-based I2V/V2V communication, in a given mobility scenario? (ii) what are the factors that mainly determine such a performance?

To answer these questions, we assume ideal conditions from a system engineering viewpoint, i.e., the availability of pre-emptive knowledge of vehicular trajectories and perfect scheduling of data transmissions, and we cast the downloading process to a mixed integer linear programming (MILP) max flow problem. The solution of such a problem yields the optimal AP deployment over a given road layout and the optimal combination of any possible I2V and V2V data transfer paradigm: it thus represents the theoretical upper bound to the downloading throughput attainable in practice.

Consider the following data transfer paradigms:

1) Direct transfers, resulting from a direct communication between an AP and a downloader. This represents the typical way mobile users interact with the infrastructure in today's wireless networks.
2) Connected forwarding, i.e. traffic relaying through one or more vehicles that create a multiple hop path between an AP and a downloader, where all the links of the connected path exist at the time of the transfer. This is the traditional approach to traffic delivery in ad hoc networks;
3) Carry-and-forward, i.e., traffic relaying through one or more vehicles that store and carry the data, eventually delivering them either to the target downloader or to another relay deemed to meet such downloader sooner.

III. PROPOSED SYSTEM

- We propose a software-defined pseudonym system with a hierarchical architecture, which leverages the SDN technology to provide flexibility and programmability for pseudonym management.

- We develop the two-sided matching theory to solve pseudonym resource scheduling problem, which matches the optimal pseudonym transmitters and receivers to decrease the system overhead due to the cost of inter-pool communications.

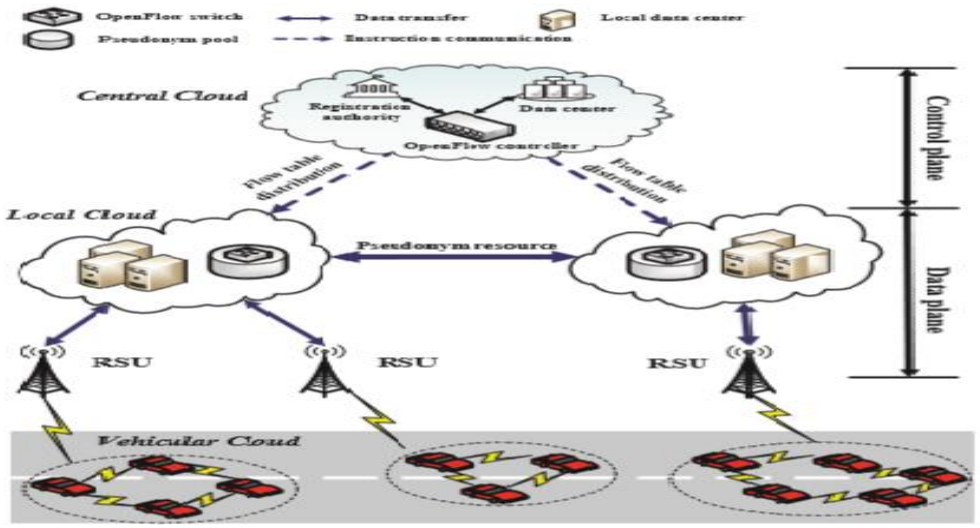


Fig. 3.1 : A ordered design of SPDS for automobiles clouds.

Advantages:

- SDN improves the pseudonym resource utilization, and effectively strengthens the vehicles' location privacy.
- SDN technology to provide flexibility and programmability for pseudonym management.

Problem Definition:

Vehicles traveling within cities and along highways are regarded as most probable candidates for a complete integration into mobile networks of the next generation. Vehicle-to infrastructure and vehicle-to-vehicle communication could indeed foster a number of new applications of notable interest and critical importance, ranging from danger warning to traffic congestion avoidance. increasing number of vehicles, pseudonym management in vehicular clouds has become a problem. heavy computing workload for the central cloud and a big backhaul delay for the vehicles.

IV. IMPLEMENTATION

Consider the urban road scenario where the vehicles provided with the communication interfaces would like to download the information while travelling with in the cities and along the highways. This will lead to some problems such as improper deployment of Access Points, reduction in the transfer rate, download rate keeps decreasing and so many other.

In order to come over from the problems mentioned above, proposed system comes with the solutions such as, a technique/method called Carry & forward fashion, proper selection of carriers, Scheduling of data chunks. With the help of Carry & forward fashion download rate can be improved. This involves two important steps are as follows:

- Production phase
- Forward phase

Here users aboard cars can exploit roadside Access Points (APs) to access the servers that host the desired contents. A minor percentage of APs is simultaneously involved in direct data transfers to downloader cars in their respective coverage area, and the majority of APs is instead idle. Make APs inactivity periods to transmit, to cars within range of idle APs, pieces of the data being currently downloaded by other vehicles. Cars that obtain information chunks this way can then transport the data in a carry& forward fashion, and deliver it to the destination vehicle, exploiting opportunistic contacts with it.

In the architecture shown above, many vehicles are travelling in a highway if any user want to download the data ('a' car), first the user has to select the data and the request is to the tower1 nearby, then that tower 'A' will calculate for how much time the vehicle will be within the transmission range of that tower and it will download that much of data. The remaining data has to be downloaded from the other vehicle ('b') which will be heading towards the download vehicle (red car),with the help of tower2 near to the other vehicle('b'). Tower 'B' calculates for how much time the

vehicle will be within the transmission range of that tower 'B' and it will download the remaining data. Once the download process is complete the remaining data will be send to the download vehicle. Finally the merging happens at the download vehicle ('a'). This is called as Carry & forward fashion.

Algorithm:

Case 1: ***Access Point A only able to Transfer the Complete File***

Step 1: Red Car requesting for File (F) to download to Access Point A
Step 2: Let x1, y1 is a Location co-ordinate for Access Point A Transmission Range end Limit and x2,y2 is a location co-ordinate for Red Car.
Step 3: Let the Distance between Access Point A and Red Car is D1.
D1 = SQRT ((x1 – x2) ^ 2 * (y1 –y2)^ 2)
Step 4: Let S1 is the Speed of the Red Car.
Step 5: Let T1 is the Time required Red Car to Cover D1 Distance
T1 = D1 / S1 * 3600
Step 6: Let T2 is the Time required to transfer the file F
Step 7: if T2 < T1 Then Transfer the file to Red Car else Call CASE 2.
Step 8: Stop.

Case 2: ***Access Point A only not able to Transfer the Complete File***

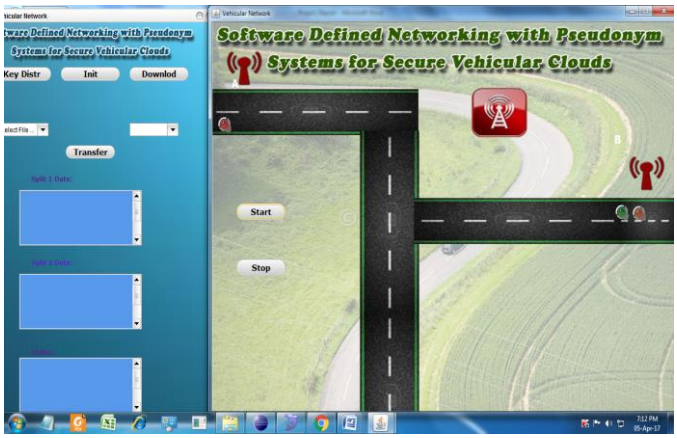
Step 1: Calculate the Bytes (B1) which can able to transfer to Red Car before it reaches the Transmission Range.
Step 2: Divide the file into two blocks BLK1 (Contains B1 bytes) and BLK2 (Contains Total bytes – B1).
Step 3: Transfer BLK2 to Access Point B
Step 4: Transfer BLK1 to Red Car.
Step 5: Once BLK2 received it has to select the Car 2 (It may either Yellow or Green).
Step 6: Transfer the BLK2 to Selected Car 2.
Step 7: Check for Red Car and Car 2 distance, If it is in transmission range goto next step (8) else wait and keep checking.
Step 8: Transfer BLK2 to Red Car.
Step 9: Merge BLK1 and BLK2 in Red Car.
Step 10: Stop.

Case 3: ***Access Point A get the request and not able to trace Red Car***

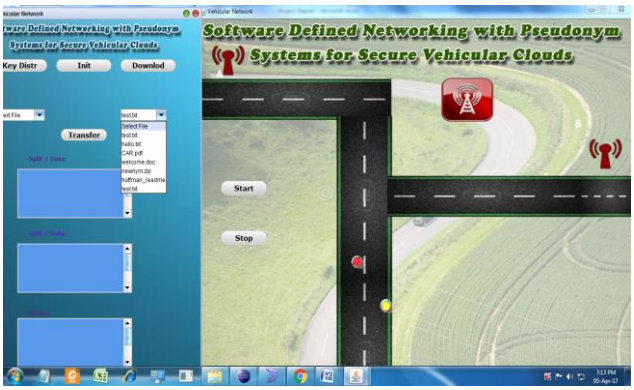
Step 1: Red Car requesting for File (F) to download to Access Point A
Step 2: Access Point A is training to get the X and Y co-ordinate of Red Car, but it is not able to trace the Red Card.
Step 3: Stop.

V. RESULTS

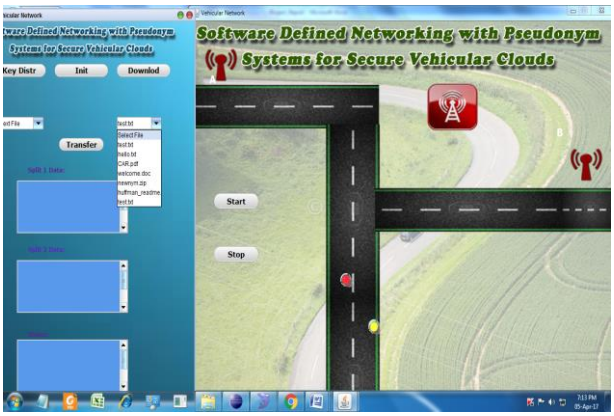
A. Initialization.



B. Folder choice.



C. Copied folder content.



VI. CONCLUSION AND FUTURE WORK

- If Downloading Vehicle is within the transmission range of Access point 1 and Requested File is small. The requested file will be completely downloaded from Access point 1 alone.
- If Downloading Vehicle is in Transmission range of Access Point 1 and Requested file is big. The Access point calculate the time that vehicle 1 presence in the transmission range. For that time period how much file can be download will be calculated and it will start download. The remaining file download is forwarded to access point 2.The Access point 2 will repeat the calculation and start down load the remaining file into the vehicle which is

heading towards vehicle 1.Once vehicle 2 reaches the vehicle 1 it start transferring the file to vehicle 1. In vehicle one it will be merged and user will get the merged file.

- If Downloading Vehicle is out of Transmission range of Access Point 1 the download request is rejected and it will not process the request. With the implementation of carry & forward transfer mechanism download rates are improved.

We can address these issues in future enhancement:

1) The system developed under single vehicle, file download processing it can be extended to multiple vehicle and the road topology can be extended.
2) VANET is not having the solution for vehicle breakdown or dynamic changing behavior of the vehicle. we can address these issues in future enhancement.

REFERENCES

[1] Dedicated Short Range Communications (DSRC), http://grouper.ieee.org/groups/scc32/dsrc/index.html.
[2] F. Bai, H. Krishnan, V. Sadekar, G. Holland, and T. Elbatt, "Towards characterizing and classifying communication-based automotive applications from a wireless networking perspective," in Proceedings of IEEE Workshop on Automotive Networking and Applications (AutoNet), pp. 1-25, 2006.
[3] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in Proceedings of the Fourth Workshop Hot Topics in Networks (HotNets-IV), Nov. 2005.
[4] S. B. Lee, G. Pan, J. S. Park, M. Gerla, and S. Lu, "Secure incentives for commercial ad dissemination in vehicular networks," in Proceedings of ACM Mobihoc, pp. 150-159, 2007.
[5] M. Raya and J. P. Hubaux, "Securing vehicular ad hoc networks," Journal of Computer Security, vol. 15, no. 1, pp. 39-68,2007.
[6] IEEE Std 1609.2-2013 - IEEE standard for wireless access in vehicular environments - Security services for applications and management messages, Apr. 2013.
[7] H. C. Hsiao, A. Studer, C. Chen, A. Perrig, F. Bai, B. Bellur, and A. Iyer, "Flooding-resilient broadcast authentication for vanets," in Proceedings of ACM Mobicom, pp. 193-204, Sep. 2011.
[8] C. Zhang, R. Lu, X. Lin, P. H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in Proceedings of IEEE INFOCOM, pp. 816-824, 2008.