IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 20-26

Emergency Message Transmission Method Based on VANET

Saurabh Patil[1], Dr. Lata Ragha
[1]*(Information Department, Terna Engineering College, India)*
[2]*(Computer Department, Fr. Agnel, vashi, India)*

Abstract: *This paper studied the problem of information transmission delay in VANET, and proposed an emergency message transmission method. The method got the messages data through netfilter architecture, reassembled the warning messages with Linux virtual device, then directed the physical network card and sent it to the destination terminal. The transmission process both reduces the time delay of traditional cache and the overhead of TCP/IP peer layers. The result shows it is effectively reducing the transmission delay during the transmission process, and improving the driving safety of the vehicle.*

Keywords: *VANET; emergency message; Netfilter architecture; virtual device*

I. Introduction

Vehicular ad-hoc network (VANET) [1,2] is ad hoc structure distributed vehicles, also distribution of messages between vehicles and roadside facilities on the road. The real network structure of the controlled short-range communication network is shown in Figure 1. In VANET there are many V2X modes, including car-to-vehicle (V2V), car-to-infrastructure (V2I), car-to-internet (V2N) and car-to-passenger (V2P). VANET technology has typical applications which includes driving safety warnings, assisted driving, distributed communication, information release or broadcast, traffic flow control information, emergency messages and many other aspects.

In the simulation safety accident experiment, according to the number of experiments, the lower the system response time, the sooner the safety warning message can be transmitted to the driver, which is equivalent to leaving the driver with more reflection time. This is to avoid the accident on road. An important goal of VANET construction is to improve the safety of vehicle driving. In order to improve the safety of vehicle driving, an important goal of the vehicle is to improve the driving safety of the vehicle by faster propagation of emergency message. This enables vehicles to share safety-related information in a timely manner. Therefore, the timeliness of sending and receiving emergency messages is high.

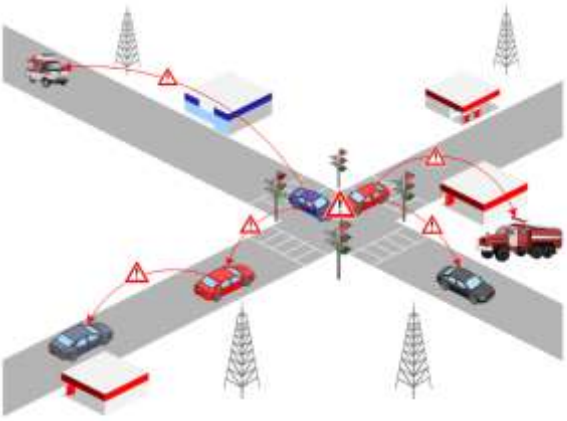

Fig. 1 VANET architecture

In recent years, there have been many studies in the literature on the vehicular self-organizing network protocol stack [3, 4] and routing algorithms [5~8]. In order to achieve efficient data routing and transmission, the paper [10] proposed to implement efficient data routing and transmission based on stoppage. Based on the data transmission strategy of the parking backbone network, the parking units are composed of virtual implementation data transmission, which has higher success rate and lower delay than the unified protocol, but the transmission success rate decreases as the vehicle generates the message rate increases. The quantity of of message and number of RSUs also has an impact on the performance of the protocol. In [11], a distance-based emergency message is proposed for the emergency message. Distance-based receiver-oriented routing protocol BDRO, the protocol has the characteristics of selective, passive and forwarding node candidate mechanisms.

The distance between the node and the source is used as the basis for selecting the forwarding. However, the transmission of the urgent message is still uses TCP/IP protocol method which can further simplify the transmission of emergency messages. The literature [12] proposes a VANET cooperative directional broadcast forwarding cooperative security early warning routing algorithm. The routing technology achieves the purpose of rapid propagation, and its idea is the same as normal message transmission during the message generation and reception phase. The existing method of sending messages is mainly the traditional TCP/IP protocol method. TCP/IP protocol adopts layered modular protocol stack design, they are mainly divided into application layer, transport layer, network layer and network interface layer. Each layer completes some functions of the packet transmission, and the data packet is from top to bottom of the protocol stack. However, in the application of the emergency message sent by the vehicular self-organizing network, the TCP/IP protocol has some shortcomings, mainly in the:

a. The transport layer and Internet layer of the TCP/IP protocol need to pack the data (add the header) before the application layer message is transmitted to the network interface layer. This may involve parsing and copying data. For emergency messages, whether it is parsing, copying, or packaging required by the transport layer and the IP layer, increases transmission delay of emergency messages.
b. For the IP layer, the IP layer mainly adds IP headers and finds the appropriate routes through different IP addresses and sends the packets to the specified terminal. In the self-organizing network of the vehicle, the safety-related emergency message can be transmitted to a certain range of vehicles around the accident vehicle, so the method of broadcasting is most suitable. And the in-vehicle network uses the 802.11p wireless communication protocol, which has an effective transmission range of 300 meters, and is sufficient for the range of vehicles to be transmitted for emergency messages. From this perspective, the IP layer function of the TCP/IP protocol is unnecessary for emergency message transmission, which only increases the transmission time of emergency messages.
c. For the transport layer, the transport layer mainly adds the transport layer header and implements connection control (reliable transmission, error control, flow control) through the transport layer header. In the vehicular self-organizing network, security-related emergency messages do not require too much application processing, and the accident vehicle will continuously broadcast emergency messages to the surroundings before the end of the accident. Therefore, the function of the transport layer is unnecessary for the transmission of emergency messages, which in turn increases the transmission delay of emergency messages.

It can be seen that the transmission layer and IP layer functions of the traditional TCP/IP protocol are not only unnecessary when transmitting security-related emergency messages in the vehicular self-organizing network, but the processing in these two layers will also increase the emergency message transmission duration.

II. Emergency Message Transmission Method

For the analysis in the previous section, the protocol model shown in Figure 2 is proposed. Figure 2 shows an emergency message processing module. The transmission process of the module reduces the traditional buffer latency and the extra overhead of the redundant IP layer and transport layer functions in the TCP/IP protocol, and preempts other transmission resources directly. The physical layer implements the purpose of sending emergency messages immediately after they are generated.

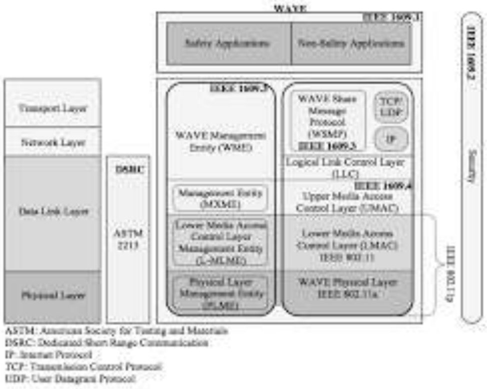
WAVE
IEEE 1609.1
Safety Applications
Non-Safety Applications
IEEE 1609.2
Transport Layer
Network Layer
Data Link Layer
Physical Layer
DSRC
ASTM
2213
IEEE 1609.3
WAVE Management Entity (WME)
WAVE Short Message Protocol (WSMP)
IEEE 1609.3
TCP/
UDP
IP
Logical Link Control Layer (LLC)
IEEE 1609.4
Management Entity (MXME)
Upper Media Access Control Layer (UMAC)
Lower Media Access Control Layer Management Entity (L-MLME)
Lower Media Access Control Layer (LMAC) IEEE 802.11
Physical Layer Management Entity (PLME)
WAVE Physical Layer IEEE 802.11a
IEEE 802.11p
Security
ASTM: American Society for Testing and Materials
DSRC: Dedicated Short Range Communication
IP: Internet Protocol
TCP: Transmission Control Protocol
UDP: User Datagram Protocol

Fig 2. Communication protocol model

According to the protocol model of Figure 2, this paper presents an emergency message transmission method. The method obtains the data information of the packet through the interception of the netfilter architecture, and reassembles the security warning information message by using the Linux virtual device, and directly points to the physical network card to send to destination receiving terminal. The block diagram of the method including the main steps as follows

1.1 Emergency message generation phase: The emergency message processing module generates an emergency message when the vehicle is in an emergency. The format of the emergency message is shown in Figure 3. In the event of an accident or other emergency in a vehicle, the system process of the emergency message processing module in the in-vehicle system preempts the CPU resource to quickly generate a corresponding emergency message, or the user process having the authority invokes the emergency message processing module to generate an emergency message.

1.2 Emergency message transmission phase: The emergency message processing module organizes and sends the generated emergency messages. The emergency message processing module organizes urgent messages to be sent, and adds necessary location information and emergency message types, such as: vehicle collision, ambulance, coordinated collision avoidance and organizes the length of emergency messages within 20 bytes. Because the effective length of the IP packet is greater than or equal to 20 bytes, the emergency packet processing module and the network layer of the TCP/IP protocol can distinguish the type of the packet by the length of the packet, and then the emergency packet processing module will handle the emergency. The message is directly delivered to the network interface layer and broadcasted to the vehicle within 300 meters of the effective range of the 802.11p wireless communication protocol.

1.3 Emergency message reception phase: The emergency message processing module receives the emergency message and processes it. According to the length information of the emergency message, the emergency message processing module determines whether the received message is an urgent message; when the network interface layer of the surrounding vehicle receives the message, it does not determine the type of the message, but separately delivers the message to the network layer and the emergency packet processing module. The network layer of the TCP/IP protocol directly discards packets with a length of less than 20 bytes and processes ordinary packets with a length of 20 bytes or longer. The emergency module directly discards packets with length greater 20 bytes and processes urgent messages with a length less than 20 bytes. When the emergency packet processing module receives the emergency packet, the corresponding system process preempts the CPU resources and preferentially processes the security-related emergency packets.

Message Type(16 bits)	**Message Priority (16 bit)**
Checksum (32 bit)	

Fig 3. Emergency Packet format

As shown in Figure 3, the message type field in the emergency message format occupies 16 bits (2 Bytes). The message type field mainly distinguishes the type of emergency affairs of the vehicle represented by the emergency message (post-accident alarm, vehicle collision avoidance). , intersections, collision avoidance, etc.), 16 bits can represent 65 535 different types. Reserved bits are used for future extended applications, and the data portion (up to 15 bytes) primarily stores location information for the vehicle. The length of the urgency message is up to 19 bytes, and the effective length of the IP packet is at least 20 Bytes. Thus, the length of the message can be used to distinguish different message types.

Therefore, this paper proposes an emergency message transmission method based on the vehicle self-organizing network, which has the following characteristics.

a. The emergency message transmission method based on the vehicle self-organizing network described is used to transmit the emergency message, so that it can bypass the header appending process of the transport layer and IP layer of the TCP/IP protocol.
b. The emergency message transmission method based on the vehicle self-organizing network described can use the special system kernel module to process the emergency message, which can conveniently preempt the CPU and save the process queue time.

Taking Changchun City as an example, it is assumed that in the event of a collision accident at the satellite Plaza in Changchun City, due to the triggering of external sensors, the emergency message generation unit of the emergency message processing module of the accident vehicle get activated. The system calls generated of the emergency message processing module to preempt CPU resources, quickly generate corresponding emergency messages, and deliver the emergency messages to the emergency message sending unit.

The emergency message sending unit adds the location information GPS of the accident vehicle to the "data" field of the emergency message, occupying 8 Bytes. Assuming that "0000000000000001" represents the type of accident of the crash, the accident type is added to the "message type" field of the emergency message occupies 2 Bytes; if there is no other important information, the "reserved bit" field of the emergency message is kept empty and not added to message, so that the emergency message occupies 10 Bytes, and then the urgent message is organized. Further processing get invokes by preempting the CPU resources and prioritize the emergency message. It is handed over to the network interface layer and broadcasted to the vehicle within the valid range (300 m) of the 802.11p wireless communication protocol.

III. Implementation Of Emergency Message Sending Mechanism

This paper uses the LAN environment to select two normal networked PCs as the development platform (sender) and test platform (receiver), and linux-gcc-3.2.25 as the overall compilation environment for the system development. Select Red Hat 9.0 Linux, all PCs in the LAN can support IPv4/IPv6 address protocol at the same time.

The structure of the emergency message sending mechanism is as shown in Figure 4. The following tasks are performed: tcpclient sending process (tcpclient) and tcp packet receiving process (tcpserver) based on the IPv6 address protocol are run in the user space. Udp packet receiving process (udpserver); two important modules in the kernel space of the Linux operating system are loaded and run normally, namely the packet module containing the netfilter architecture and the vdev module containing the Linux virtual device, both The technology comes from Linux device drivers. These programs can be combined to generate VANET emergency messages on the Linux-based mobile high-speed mobile terminal and send them out immediately after the generation.

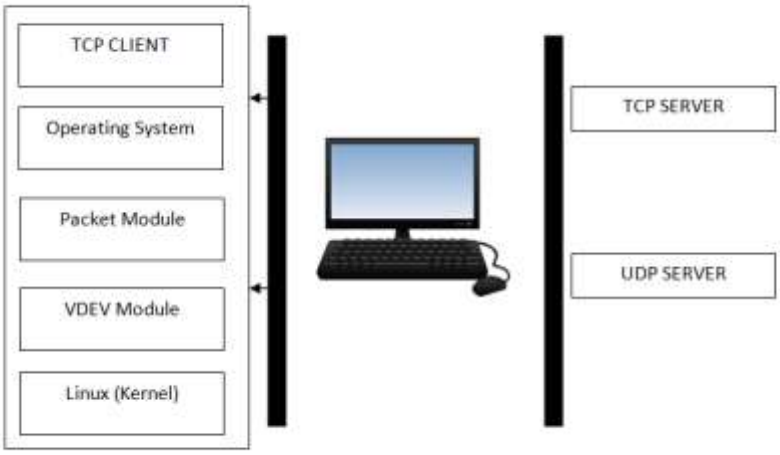

Figure 4 Schematic diagram of VANET emergency message sending mechanism

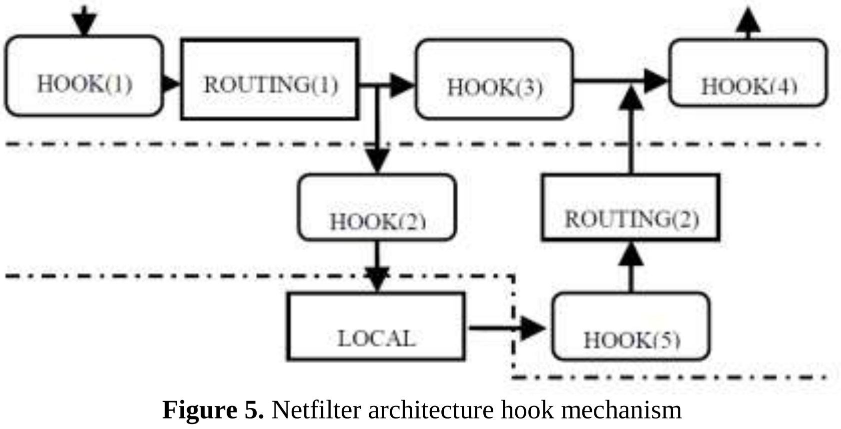

Figure 5. Netfilter architecture hook mechanism

1.4 Netfilter Architecture

The netfilter that appears after the Linux 2.4 kernel is a firewall architecture. To achieve status detection, packet filtering and other functions. It can use its open interface to implement its own functional modules in the corresponding protocol layer. The netfilter architecture defines five hook (HOOK) functions for the IPv4 and IPv6 address protocols in the network protocol, as shown in Figure 5, in the IPv6 protocol:

HOOK(1), NF_IP6_PRE_ROUTING;
HOOK(2), NF_IP6_LOCAL_IN;
HOOK(3), NF_IP6_FORWARD;

HOOK(4), NF_IP6_POST_ROUTING;
HOOK(5), NF_IP6_LOCAL_OUT.

The specific data processing flow of the netfilter architecture is explained. When various data packets transmitted in the network are to enter the system from the data link layer, the IP address verification is first performed, and the first hook HOOK(1) (NF_IP6_PRE_ROUTING) get started. its role is to determines whether the packet is at destination or need to be forwarded. If it is forwarded, it will be caught and forwarded by the hook HOOK(3) (NF_IP6_FORWARD); if it is destination machine, it will be caught by the hook HOOK(2) (NF_IP6_LOCAL_IN), and pass to the upper layer protocol. The forwarded packet will be caught and processed by the 4th hook HOOK(4) (NF_IP6_POST_ROUTING). This packets are then processed by the hook HOOK(5) (NF_IP6_LOCAL_OUT), and finally processed for performing operation such as routing, and again processed by the 4th hook HOOK(4) (NF_IP6_POST_ROUTING) at the destination. After that, it is transferred to the network layer again. Each hook function in the protocol, after processing is completed, it return a value as an integer constant. The corresponding processing of the data packet by the Linux kernel is based on these return values. Specifically, these return values include:
a) NF_DROP: discard without any processing
b) NF_ACCEPT: Receive for the next step;
c) NF_STOLEN: abnormal grouping;
d) NF_REPEAT: re-enter this hook;
e) NF_QUEUE: Enter into user space queue, waiting for the corresponding process to process.

The packet module in the VANET emergency message sending mechanism mainly uses the NF_IP6_LOCAL_OUT HOOK (hook function) in the five HOOK (hook functions) in the netfilter architecture to implement the data.

When the packet is sent out from the source process, it is captured and then the corresponding connection tracking and packet processing operations is performed. The specific workflow of the netfilter architecture technology can be shown in Figure 6.

1.5 Linux device driver technology

In the Linux operating system, everything is a file. The actual hardware device corresponds to a "device file". In summary, there are three main types of these devices: character devices, block devices, and networks. Interface, and independent of the kernel, can be inserted when necessary, and deleted when not necessary.

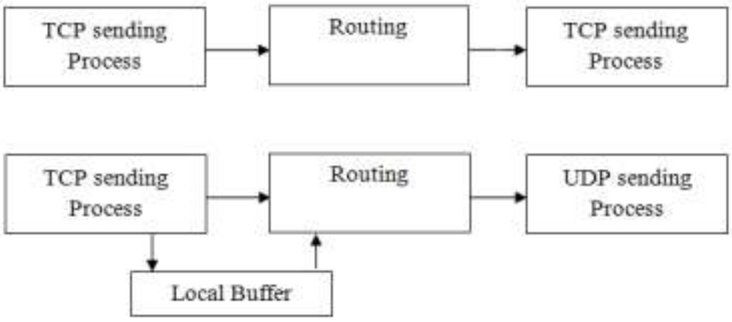

Figure 6.net tfilter architecture workflow

The structure of the Linux device driver main application technique in the VANET emergency message sending mechanism studied in this paper is the driver technology of the network device part in the Linux device driver, namely the network interface. The Linux device driver module used in this paper is the "vdev" module. The module mainly implements three functions: obtaining useful information in TCP packets from the "packet" module containing the netfilter architecture technology; The information to be sent is extracted and reassembled into a UDP emergency message that meets the requirements; the newly assembled UDP message is sent directly to the network card "eth0" without going through the protocol stack and various cache queues.

1.6 VANET emergency message sending mechanism implementation process

The VANET emergency message sending mechanism utilizes socket network programming technology, netfilter architecture technology and Linux device driver technology under. It can be seen that the location of the emergency message sending mechanism running in the computer architecture according to the program involved can be divided into three parts, the first part is a process program running in the user space;

the second part is used to manage the user space and Linux kernel space and the protocol stack related to network information transmission; the third part is mainly for the mechanism to accelerate the message sending closely related to the two packet module and vdev module of kernel. According to the whole network communication process, it can be divided into two parts. The first part is the sending terminal part responsible for generating and transmitting VANET emergency messages; the other part is about receiving terminal responsible for handling emergency messages and verifying the normal operation of the entire VANET mechanism. A transmitting terminal, a receiving terminal and a LAN environment through which the entire information transmission process passes together constitute a network information transmission test platform.

The first part is at the highest level of the entire network protocol and is primarily responsible for fulfilling the specific requirements of the user. In the overall development environment of the mechanism, it involves the sending terminal responsible for generating and transmitting emergency messages, and the receiving terminal responsible for receiving emergency messages and testing the entire emergency message sending mechanism.Among the three most important key technologies, it uses the socket network programming technology under Linux [9], for overall process of transmission of the entire network information. The specific disassembly can be divided into three independent processes, which are the TCP packet sending process (tcpclient) running on the sending terminal to implement TCP emergency packet generation and sending, and running in the receiving terminal to achieve interception through the netfilter architecture. Any TCP packet receiving process (tcpserver) that processes TCP emergency packets again, runs on the receiving terminal and is captured and processed by the netfilter architecture, and then reassembled into UDP emergency packets in the virtual device, and finally by the virtual device. The UDP packet receiving process (udpserver) that directly points to the UDP emergency packet sent by the NIC (eth0).

The second part is the Linux operating system for managing user space and kernel space and the protocol stack associated with network information transfer. There is no modification in the emergency message delivery mechanism proposed in this paper, or the TCP/IP protocol stack used.

The third part, the two kernel modules running in the kernel space and the mechanism to accelerate the packet sending method are the packet module including the netfilter architecture technology under Linux and the vdev module implementing the virtual device technology. In the packet module, the main application technology is the netfilter architecture technology under Linux. First, load the packet module into the Linux kernel with the insmod command. Then, use the hook (HOOK) function provided by the netfilter architecture (NFOK) function NF_IP6_LOCAL_OUT to capture the relevant TCP message, then store it in the structure sk_buff, and then apply the pointer to get The data information data to be transmitted, and the data information data is passed to the vdev module, and the return value of the entire netfilter architecture is NF_ACCEPT, which means that the captured TCP emergency message is returned to the captured location as it is, and continues to follow the traditional transmission mechanism. The transmission is performed, indicating that two types of emergency messages, TCP format and UDP format, containing the same information are sent from the transmitting terminal. After all tasks are completed, the rmmod command can be executed to unload the packet module. This packet module performs the two main functions that it is responsible for for obtaining message information and for passing emergency information to the vdev module. The vdev module is first loaded into the kernel. Use the lsmod command to determine whether the lsmod command is successfully used. Then, the function vdev_module_rx(data) in the vdev module is called by the packet module to transfer the useful information in the TCP emergency packet obtained by the packet module to vdev. The module, the vdev module re-groups the obtained useful information into a UDP emergency message through the function vdev_module_rx, and then calls the function vdev_xmit in the function vdev_module_rx to directly send the reassembled UDP message directly to the network card (eth0). Once the task is complete, the vdev module can be uninstalled by the rmmod command. After the above process, the vdev module completes its three main functions of obtaining useful information from the packet module, reassembling the useful information into UDP emergency messages, and finally directly sending the eth0.

This paper implements the VANET emergency message sending method test program, which works in a LAN range, and the IPv6 protocol stack is configured between two hosts communicating with each other in the network. The sending terminal only runs a TCP packet sending process (tcpclient), which will send emergency message related to the driving safety problem stored in the text file is generated and sent out, and the TCP packet receiving process (tcp server) responsible for receiving the TCP emergency message and the UDP receiving the UDP emergency message are simultaneously operated at the receiving terminal. Message receiving process receives a message containing the same urgent information. The test results show that both the TCP emergency packet receiving process and the UDP emergency packet receiving process have received emergency message information, indicating that the VANET emergency packet is studied and implemented in this paper.The delivery mechanism accurately completes the function of sending emergency messages.In addition, during the actual operation, the UDP emergency packet receiving process can receive packets faster than the TCP emergency packet receiving process, which can effectively reduce the transmission delay time of emergency packets.

IV. Conclusion

The VANET Emergency Message Delivery Mechanism is a new field of research involving many aspects of communication technology, network technology, computer technology and transportation technology. This paper deeply studies and analyzes the message sending mechanism, and gives a VANET emergency message transmission method. The method intercepts the data information of the packet through the hook function of the netfilter architecture, and then reassembles the security warning information message by using the Linux virtual device, and directly points to the physical network card to send to the destination receiving terminal. The method can overcome the packaging process of the transport layer and the IP layer of the TCP/IP protocol when transmitting the emergency message of the on-board self-organizing network; and simultaneously processing the emergency message by using the special system kernel module can conveniently seize the CPU and save the process time of queuing. Compared with the route forwarding protocol [12], the emergency message transmission mechanism quickly generates corresponding emergency messages in the event of an accident or other emergency of the vehicle, or the emergency message is called by the user process having the authority. The processing module generates an emergency message; in the sending phase, the emergency message is controlled to be delivered to the network interface layer for broadcasting within 20 bytes; in the receiving phase, the network layer and the emergency packet processing module of the TCP/IP protocol are used as emergency packets. When the processing module receives an emergency packet, the corresponding system process preempts the CPU resources and preferentially processes security-related emergency packets.

The main contributions of this paper are:

a) The solution and implementation of the relatively complete VANET emergency message sending mechanism is established;

b) The appropriate software and hardware environment is selected to develop the emergency message sending mechanism. Linux of Red Hat 9.0 was selected as the operating system and pure IPv6 test environment of the development platform;

c) The high-level application development applied to the VANET emergency message sending mechanism was completed by using socket network programming technology;

d) An emergency report was designed. The text format can effectively distinguish between emergency messages and common messages, and prepares for the VANET emergency message sending mechanism.

e) Using the netfilter architecture technology under Linux to implement emergency message information in the VANET emergency message sending mechanism. Capture;

f) Use the netfilter architecture technology to process the emergency message information and reassemble it into a UDP emergency message, and directly send it to the network card for fast forwarding.

References

[1] Rene O, Carlos M, Azzedine B, et al. Reliable data dissemination protocol for VANET traffic safety applications [J]. Ad hoc Networks, 2017, 63: 30- 44.

[2] Jun Tao, Yifan Xu, Ziyi Zhang, et al. A resource allocation game with restriction mechanism in VANET cloud [J]. Concurrency & Computation Practice & Experience, 2016, 29 (14).

[3] Barakat P M, Tarek R S, Khaled S. Performance study of manet routing protocols in VANET [J]. Arabian Journal for Science & Engineering, 2016:

[4] Tariq E, Layth A K A D, Yamaan M. Review and performance comparison of VANET protocols: AODV, DSR, OLSR, DYMO, DSDV & ZRP [C]// Proc of AIC-MITCSA. 2016: 1-6.

[5] Raj K J, Jaidhar C D. Location prediction algorithm for a nonlinear vehicular movement in VANET using extended Kalman filter [J]. Wireless Networks, 2016: 1-16.

[6] Lochert C, Hartenstein H, Tian J, et al. A routing strategy for vehicular ad hoc networks in city environments [C]// Proc of IEEE Intelligent Vehicles Symposium. 2003, 156-161.

[7] Sharma H L, Agrawal P, Kshirsagar R V. Multipath reliable range node selection distance vector routing for VANETT: design approach Proc of International Conference on Electronic Systems, Signal Processing and Computing Technologies. 2014: 280-283.

[8] Feng Huifang, Liu Chunfeng, Shu Yantai, et al. Location Prediction of Vehicles in VANET Using A Kalman Filter [J]. Wireless Personal Communications, 2015, 80 (2): 543-559.

[9] Yang Qiuli, Jin Zhi. Windows Network Programming [M]. 2nd edition. Beijing: People's Posts and Telecommunications Press, 2015.

[10] Zhu Jinqi, Ma Chunmei, Liu Ming, et al. Number of parking backbone networks based on vehicle self-organizing network According to Transmission [J]. Journal of Software, 2016, 27 (2): 432-450.

[11] Feng Shuo. Research on Emergency Message Routing Mechanism in VANET [D]. Changchun: Jilin University, 2014

[12] Xu Bo, Xu Shouzhi, Guo Pengfei, et al. Traffic safety warning based on shortest delay priority VANET Routing Protocol [J]. Journal of Huazhong University of Science and Technology: Natural Science Edition. 2013, 41 (S2): 242-246.