Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 6, July 2021: 6617-6629

Analysis of problems in big data transfer using SOME/IP-TP

Yujin Choi [a], Seung Jun Lee [b], Yoon Sang Kim [c]

[a] BioComputing Lab, Department of Computer Science and Engineering,
Korea University of Technology and Education (KOREATECH),
Cheonan, Republic of Korea (ekgkrha@koreatech.ac.kr)
[b] AirPlug, Republic of Korea (sjlee0123kr@gmail.com)
[c] BioComputing Lab, Institute for Bio-engineering Application Technology,
Department of Computer Science and Engineering, KOREATECH,
Cheonan, Republic of Korea (yoonsang@koreatech.ac.kr)

Abstract

Rapid advances in automotive applications led to need for big data transmission in vehicles. For these requirements, SOME/IP Transport Protocol(SOME/IP-TP) was proposed in SOME/IP(Scalable service-Oriented MiddlewarE over IP), which is a service-oriented vehicle communication. SOME/IP-TP is a method to segment a big data into lots of small-scaled smaller data. This makes it possible to transmit big data using UDP. In this paper, we analyzed the problem caused by connectionless and unreliable characteristics of UDP when using SOME/IP-TP. First, the abnormal transmission process of data due to each characteristic was investigated. Second, exceptions were defined as Shutdown and Missing, respectively. Finally, each exception was analyzed for message type transmitted through SOME/IP-TP. From the results, it was confirmed that unnecessary data transmission problems occurred when Shutdown of the Response message receiver or Missing of segmented data

Keywords: SOME/IP-TP, SOME/IP, UDP, Autonomous Vehicle

1. Introduction

An autonomous vehicle refers to a vehicle which can recognize the driving environment by itself and drive to a target point without manipulation by the driver[1, 2]. Recently, as the autonomous vehicle is noticed, various automotive applications such as Infotainment[3, 4] and advanced driver assistant system(ADAS)[5, 6] are studied actively. Common requirements of these automotive applications are the wider bandwidth and the transmission of big data.

Conventional vehicle communication technologies such as controller area network(CAN)[7], FlexRay[8], local interconnect network(LIN)[9], media oriented systems transport(MOST)[10] are difficult to achieve the above requirements, which thus lead to the introduction of Ethernet[11] in the vehicle. Because Ethernet provided wider bandwidth than the above-mentioned conventional ones[12, 13], it was able to support the increasing demand for the number and capacity of ECUs[13, 14, 15] which is exponentially increased.

Scalable service-Oriented MiddlewarE over IP(SOME/IP)[16] is one of a vehicle communication technologies based on Ethernet. Conventional SOME/IP used transmission control protocol(TCP) as a transport protocol for big data transmission[17]. However, TCP is not suitable for an vehicle environment requiring fast data, since it is difficult to guarantee a real-time performance[18, 19]. To supplement it, SOME/IP-TP[16], a method of transmitting big data using user datagram protocol(UDP), was proposed.

SOME/IP-TP is a method to segment big data into segments which is smaller than maximum transmission unit(MTU) and this makes it possible to transmit data bigger than MTU through UDP. UDP, the transport protocol of SOME/IP-TP, has an advantage in terms of real-time performance compared to TCP[18, 20]. However, UDP has the following characteristics[21, 22, 23]: 1)connectionless, in which connections between communication peers are not explicitly established or terminated. 2) unreliable, in which data packets can be lost or received in a different order. Therefore, current SOME/IP-TP has limitations that does not consider the problem due to characteristics of UDP.

In this paper, we will analyze the problems that may occur when transferring big data using SOME/IP-TP due to the above-mentioned limitations (which is caused by the characteristics of UDP such as connectionless and unreliable). Problems are analyzed as the following steps. First, the abnormal transmission process (hereafter, exception) when transmitting big data using SOME/IP-TP due to each characteristic of UDP is defined. Second, the defined exceptions are analyzed for message type transmitted through SOME/IP-TP.

The paper is composed of five sections. In section 2, we introduce SOME/IP-TP. In section 3, we define SOME/IP-TP exceptions due to each characteristic of UDP. In section 4, we analyze each exception per message type that can be transmitted through SOME/IP-TP. Finally, in section 5, we present conclusions and future works.

2. SOME/IP-TP

In this section, we introduce SOME/IP-TP, which is the subject of this study. For this, at first the environment in which SOME/IP-TP operates is described.

SOME/IP-TP is a method of transmitting big data in SOME/IP, so it is operated in an environment including SOME/IP stack. ECU(device) structure including the SOME/IP stack is represented as shown in Fig. 1.

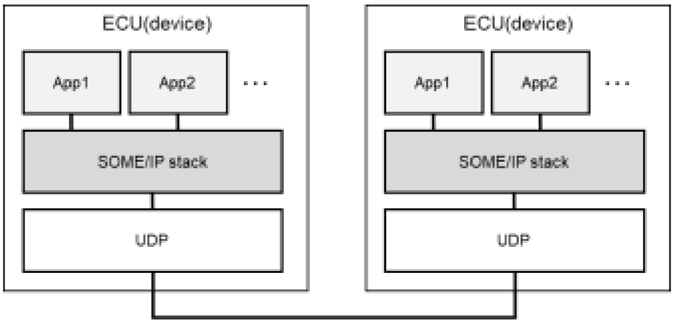

Figure 1. ECU(device) structure with SOME/IP stack

ECU structure including SOME/IP stack is described in Fig. 1. In one ECU, there is a SOME/IP stack, and the SOME/IP stack communicates with SOME/IP stack in other ECU through UDP. The SOME/IP stack is connected to several applications in the ECU and each application can perform role as Provider, Consumer or both.

The environment in which SOME/IP-TP is operated is briefly described in above. Next, SOME/IP-TP operating in the introduced environment will be described in more detail.

When transmitting data in SOME/IP using UDP, only the data smaller than MTU can be transmitted. If the data bigger than MTU should be transmitted through UDP, SOME/IP-TP should be used[16]. Fig. 2 shows an example in which SOME/IP message containing big data is transmitted from Provider application to Consumer application using SOME/IP-TP.

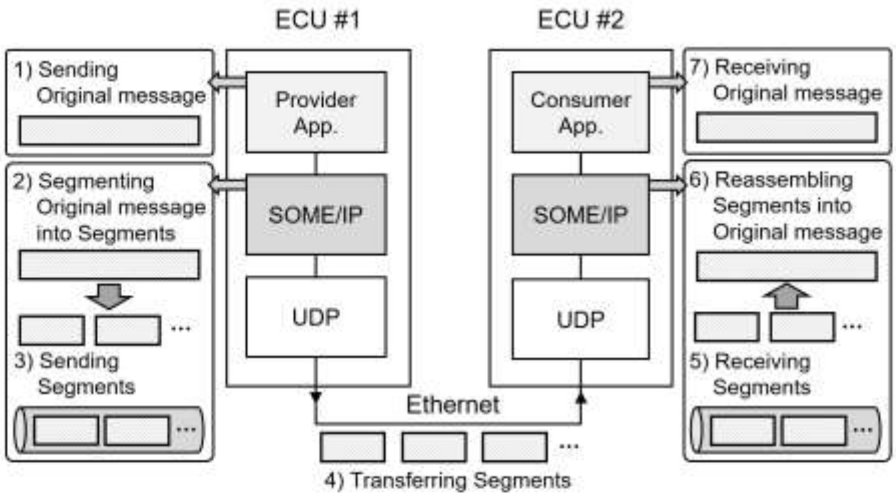

Figure 2. SOME/IP-TP message transmission process

In Fig. 2, original message means SOME/IP message containing too big data to transmit directly using UDP. Segment means each parts of original message payload transmitted through SOME/IP-TP. When transmitting big data using SOME/IP-TP, original message which is not fit to one UDP packet is segmented into segment. Each segment is transmitted from sender to receiver. Receiver reassembles completely received segments into one original message and then sending it to the final destination application. SOME/IP-TP message format using in this is shown in Fig. 3. And, the 'Message Type' field of SOME/IP-TP is shown in Fig. 4.

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="32">Message ID (Service ID / Method ID) [32 bit]</td></tr>
<tr><td colspan="32">Length [32 bit]</td></tr>
<tr><td colspan="32">Request ID (Client ID / Session ID) [32 bit]</td></tr>
<tr><td colspan="8">Protocol Version [8 bit]</td><td colspan="8">Interface Version [8 bit]</td><td colspan="8">Message Type [8 bit]</td><td colspan="8">Return Code [8 bit]</td></tr>
<tr><td colspan="28">Offset [28 bit]</td><td colspan="3">Reserved Flags [3 bit]</td><td>M [1 bit]</td></tr>
<tr><td colspan="32">Payload [variable size]</td></tr>
</table>

Figure 3. The field format of SOME/IP-TP message

<table>
<tr><td colspan="8">Message Type [8 bit]</td></tr>
<tr><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td></tr>
<tr><td>x</td><td>x</td><td>0/1</td><td>x</td><td>0/1</td><td>x</td><td>0/1</td><td>0/1</td></tr>
<tr><td>-</td><td>-</td><td>TP Flag</td><td>-</td><td colspan="4">Message Usage</td></tr>
</table>

Figure 4. The field format for 'Message Type' of SOME/IP-TP

The third highest bit of the ‘Message Type’ field is called TP Flag. When the SOME/IP message is a segment, TP Flag is set to 1. Also, the lower 4 bits of the ‘Message Type’ field are set to a predefined value based on Message Type. SOME/IP-TP ‘Message Type’s according to these rules are shown in Table 1.

Table 1. ‘Message Type’s supported in SOME/IP-TP

Message Type	Number	Description
TP_REQUEST	0x20	A TP request expecting a response
TP_REQUEST_NO_RETURN	0x21	A TP fire&forget request
TP_NOTIFICATION	0x22	A TP request of a notification/event callback expecting no response
TP_RESPONSE	0xA0	The TP response message
TP_ERROR	0xA1	The TP response containing an error

Among the ‘Message Type’ in Table 1, the contents of TP_REQUEST and TP_REQUEST_NO_RETURN are different but the purposes are the same as Request. Similarly, the contents of TP_RESPONSE and TP_ERROR are different but the purposes are the same as Response. In addition, in these cases, the message transmission process between the Provider and the Consumer is the same when transmitting big data using SOME/IP-TP. Therefore, the SOME/IP-TP message transmission process will be described as a representative of the following three types: Notification, Request (representing Request and Request no return), Response (representing Response and Error). The conventional SOME/IP-TP message transmission process is shown in Fig. 5, 6, and 7.

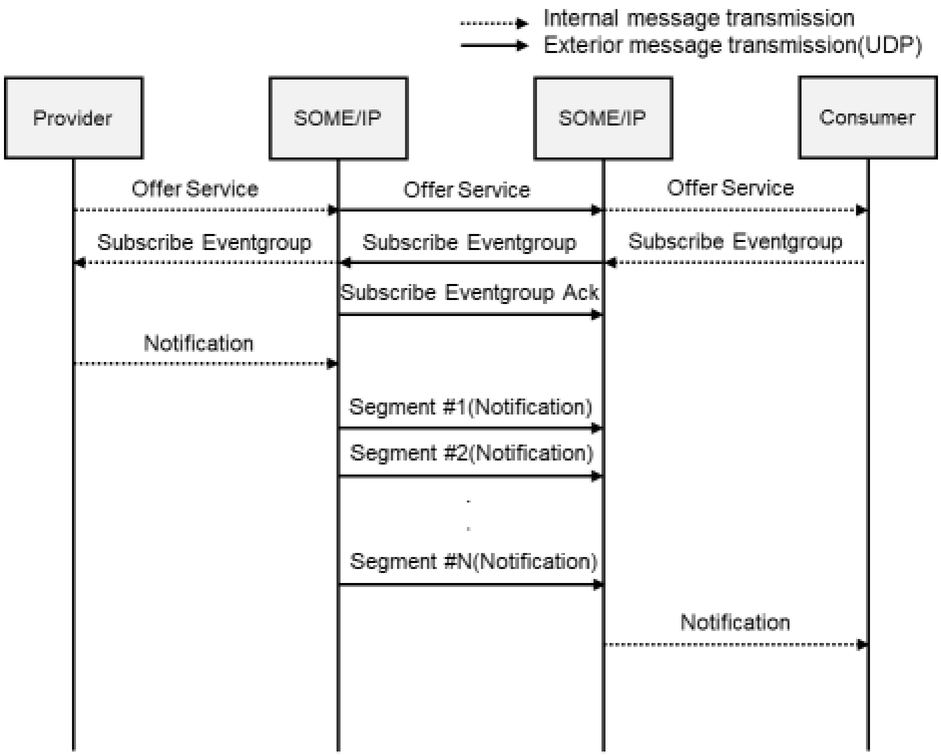

Figure 5. SOME/IP-TP message transmission (I): Notification case

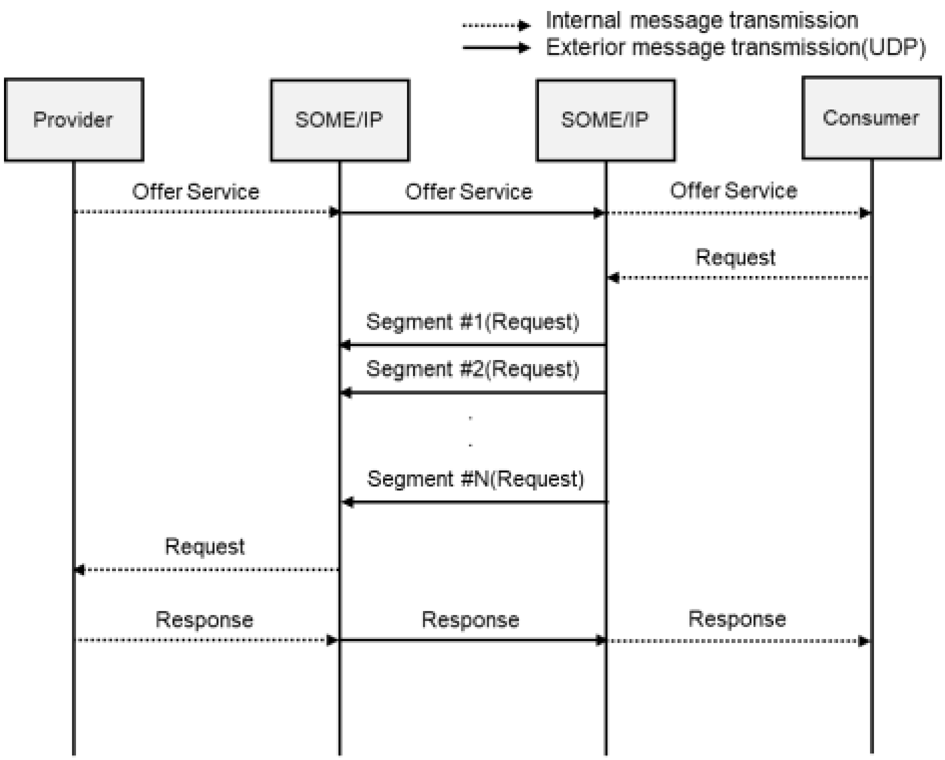

Figure 6. SOME/IP-TP message transmission (II): Request case

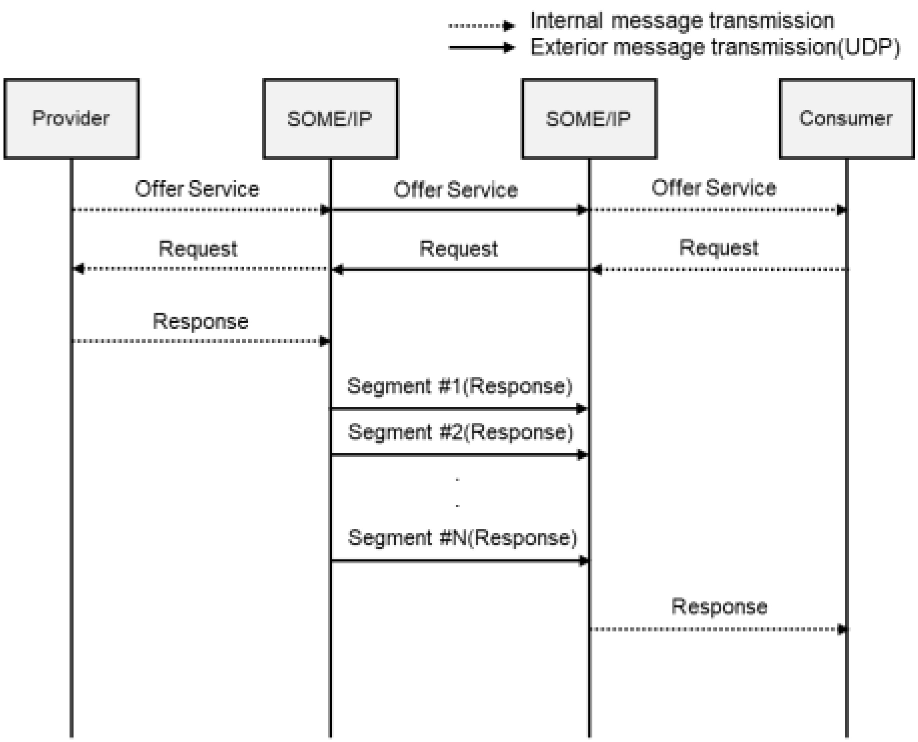

Figure 7. SOME/IP-TP message transmission (III): Response case

3. Problem Statements: SOME/IP-TP Exceptions

UDP has a characteristic such as connectionless, in which connections between communication peers are not explicitly established or terminated. Also, the UDP has a characteristic such as unreliable, in which data packets can be lost or received in a different order[21, 22, 23]. Thus, these can cause exception in SOME/IP-TP that a message is not transmitted correctly, besides the normal message transmitting process shown in Fig. 5, 6 and 7.

UDP has connectionless characteristics so that even if a communication peer is shutdown without any notice, there is no method for knowing that state. A method of transmitting a specific predefined message (Stop Offer Service, Stop Subscribe Eventgroup) is introduced in the transmission of a typical SOME/IP message when shutdown of the sender or receiver. However, it is not considered about shutdown of the sender or receiver in the transmission of SOME/IP-TP message. This is an exceptional case that can cause some problems when transmitting data through SOME/IP-TP. In this paper, this is defined as 'Shutdown'. In other words, 'Shutdown' means an exception that sender or receiver is shutdown without any notice.

Meanwhile, UDP has unreliable characteristics so that even if the message in different order is received, there is no method to know exactly whether some messages were lost or the message was received in a different order. In SOME/IP-TP, this disadvantage can be improved by a method to maintain receiving buffer until all segments arrive. However, this method is difficult for applying to an actual system in terms of memory management. For this reason, in SOME/IP-TP, 'Reorder distance' is introduced to distinguish whether some segments were lost or the segment was received in a different order.

'Reorder distance' means the threshold of the difference between the order of segment expected to be received and the order of segment actually received. In the case that the difference between the order of segments (which is expected to arrive) and the ones (which is actually received) exceeds 'Reorder distance', the receiver of SOME/IP-TP message judges that some segments were lost rather, not received in wrong order. Once the receiver of SOME/IP-TP judges that some segments were lost, all received segments were discarded and then all the following ones for the same original message were ignored. This case is also an exceptional case like 'Shutdown' that can cause some problems when transmitting data through SOME/IP-TP. In this paper, this is defined as 'Missing'. In other words, 'Missing' means an exception that some of the segments to be received are lost during transmitting segments through SOME/IP-TP continuously.

4. Analysis Results

In this section, we analyze each exception defined in section 4, per message type that can be transmitted through SOME/IP-TP. For analysis of exception, as in section 2, we assume the following three cases: Request, Response, Notification.

SOME/IP-TP message transmission process when the Shutdown occurs in the sender is shown in Fig. 8, 9 and 10.

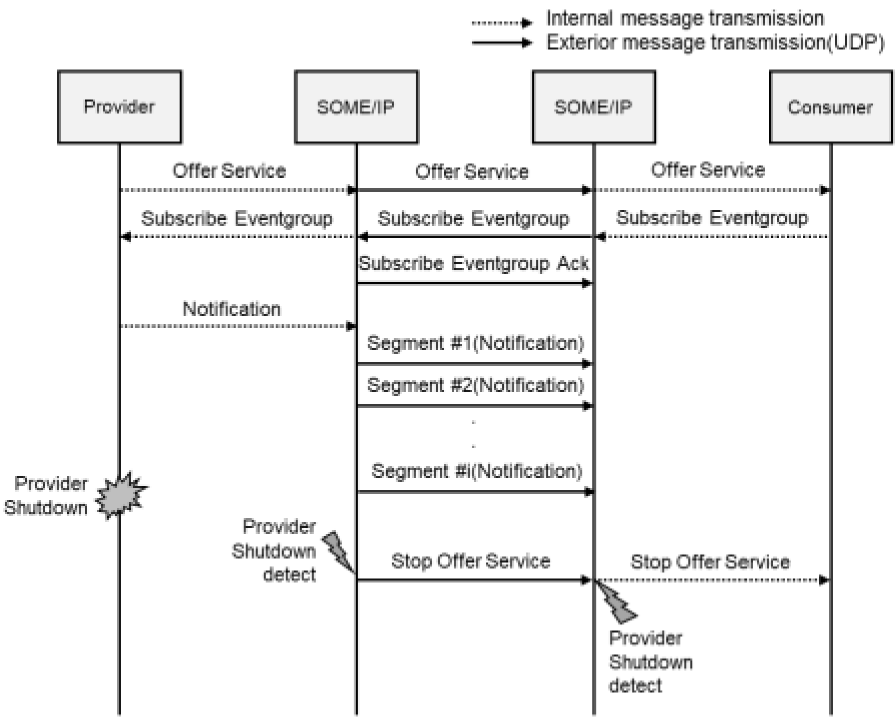

Figure 8. Problem case of SOME/IP-TP message transmission, Shutdown in the sender (I): Notification case

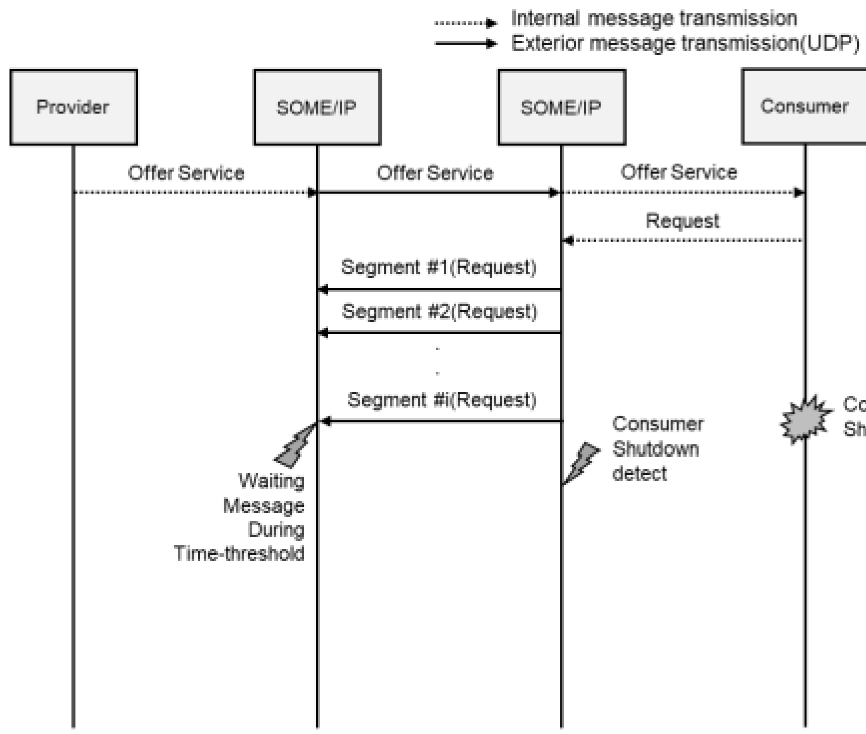

Figure 9. Problem case of SOME/IP-TP message transmission, Shutdown in the sender(II): Request case

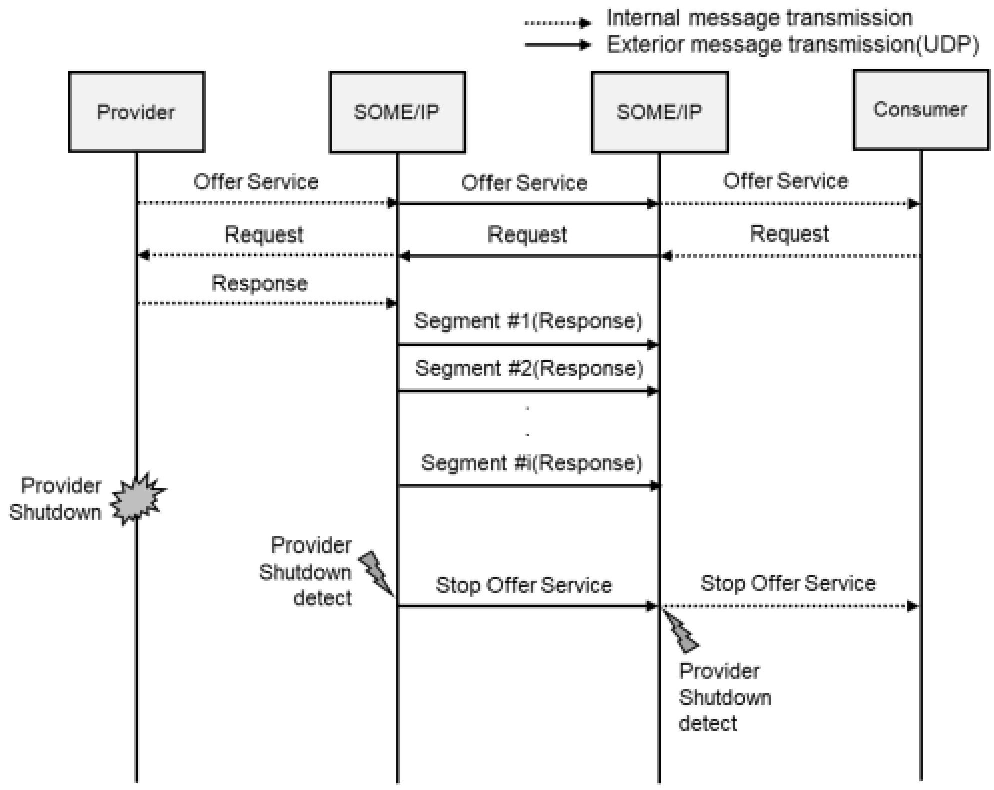

Figure 10. Problem case of SOME/IP-TP message transmission, Shutdown in the sender(III): Response

As shown in Fig. 8 and 10, the sender is a Provider when the type of SOME/IP-TP message is Notification or Response. When Shutdown of Provider occurs, it is detected by SOME/IP stack in the same ECU. After detecting the Shutdown, the SOME/IP stack informs that data offering is stopped by sending Stop Offer Service message. This makes the Consumer can detect that the Provider sending Notification or Response to itself is in Shutdown. After detecting the Shutdown of Provider, the Consumer notices that there will be no longer

Notification or Response sending from the Provider. And accordingly, the Consumer discards all of the previous segments.

As shown in Fig. 9, the sender is a Consumer when the type of SOME/IP-TP message is Request. When Shutdown of Consumer occurs, it is detected by SOME/IP stack in the same ECU. However, since the Consumer is not subscribed to the Provider, SOME/IP stack does not take any action after detecting Shutdown. For this reason, the Provider can't detect that the Consumer sending Request to itself is in Shutdown. In this case, the Provider waits for the next segment for the set waiting time. If the next segment was not received within the set waiting time, it is confirmed as a timeout and all the previously received segment are discarded.

SOME/IP-TP message transmission process when the Shutdown occurs in the receiver is shown in Fig. 11, 12 and 13.

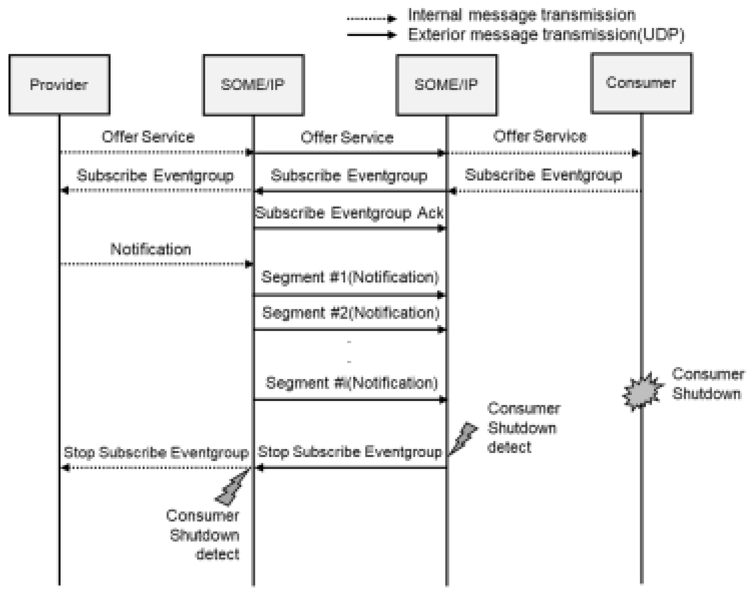

Figure 11. Problem case of SOME/IP-TP message transmission, Shutdown in the receiver: Notification case

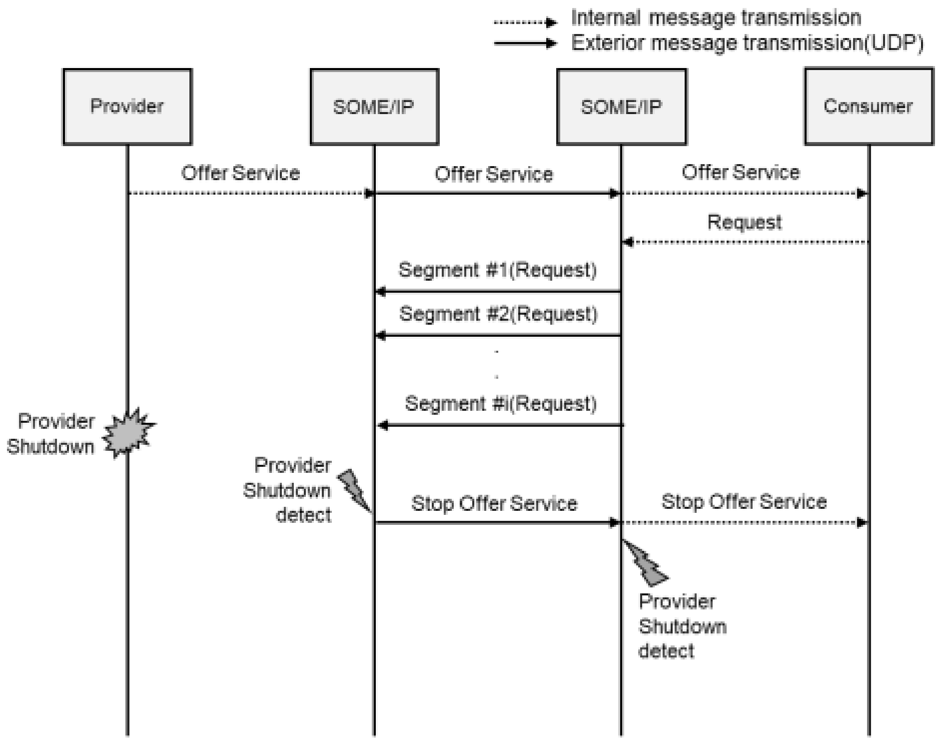

Figure 12. Problem case of SOME/IP-TP message transmission, Shutdown in the receiver: Request case

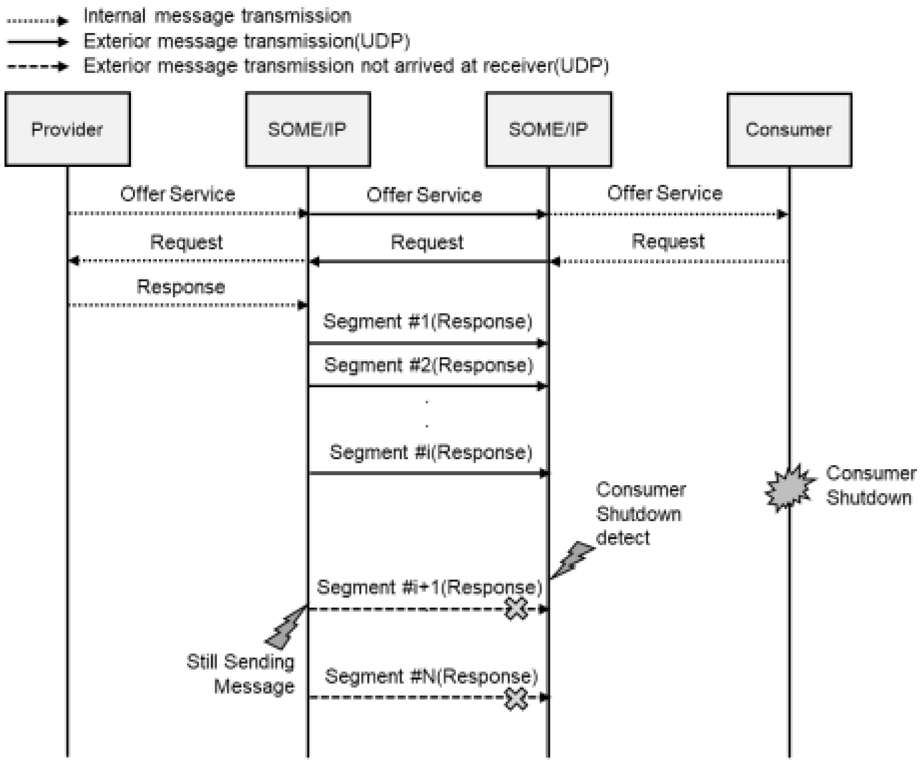

Figure 13. Problem case of SOME/IP-TP message transmission, Shutdown in the receiver: Response case

First, as shown in Fig. 11, the receiver is a Consumer when the type of SOME/IP-TP message is Notification. For receiving Notification, the Consumer is subscribed to the Provider. When Shutdown of Consumer occurs, it is detected by SOME/IP stack in the same ECU. After detecting the Shutdown, the SOME/IP stack informs that subscription is stopped by sending Stop Subscribe Eventgroup message. This makes the Provider can detect that the Consumer receiving Notification from itself is in Shutdown. After detecting the Shutdown of Consumer, the Provider notices that the Consumer can’t receive Notification. And accordingly, the Provider doesn’t send segments any longer.

Next, as shown in Fig. 12, the receiver is Provider when the type of SOME/IP-TP message is Request. When Shutdown of Provider occurs, it is detected by SOME/IP stack in the same ECU. After detecting the Shutdown, the SOME/IP stack informs that data offering is stopped by sending Stop Offer Service message. This makes the Consumer can detect that the Provider receiving Request from itself is in Shutdown. After detecting the Shutdown of Provider, the Consumer notices that the Provider can’t receive Request. And accordingly, the Consumer doesn’t send segments any longer.

Finally, as shown in Fig. 13, the receiver is Consumer like Notification, when the type of SOME/IP-TP message is Response. When Shutdown of Consumer occurs, it is detected by SOME/IP stack in the same ECU. However, since the Consumer is not subscribed to the Provider, SOME/IP stack does not take any action after detecting Shutdown. In Fig. 13, the Consumer is not subscribed to the Provider, so before Shutdown, there is no action. For this reason, the Provider can’t detect that the Consumer receiving Response from itself is in Shutdown. In this case, a problem happens the Provider continuously sends segments that can't be reached to Consumer(segments #i~#N in Fig. 13). This wasted data transmission is a problem that makes the unnecessary load on the Provider. Also, this is a problem that unnecessarily occupies bandwidth on the network.

Finally, SOME/IP-TP message transmission process when the Missing occurs is shown in Fig. 14, 15 and 16.

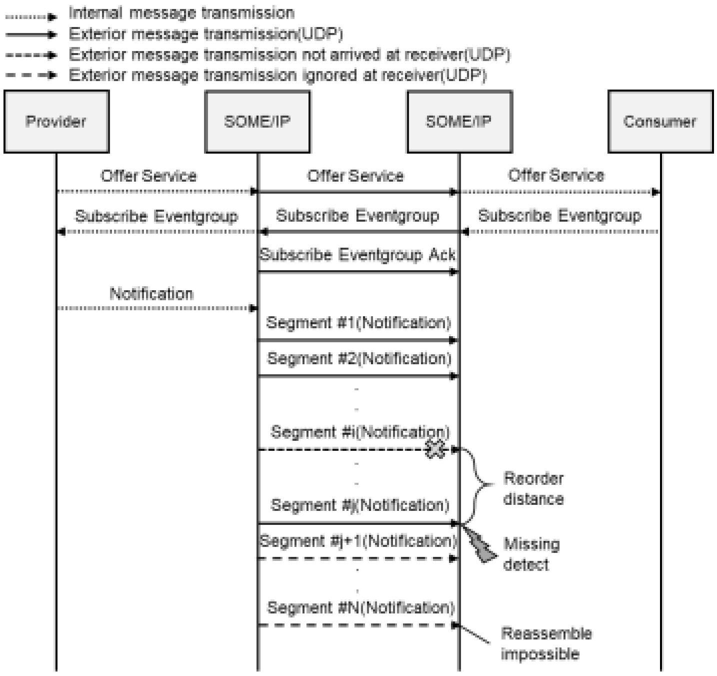

Figure 14. Problem case of SOME/IP-TP message transmission, Missing: Notification case

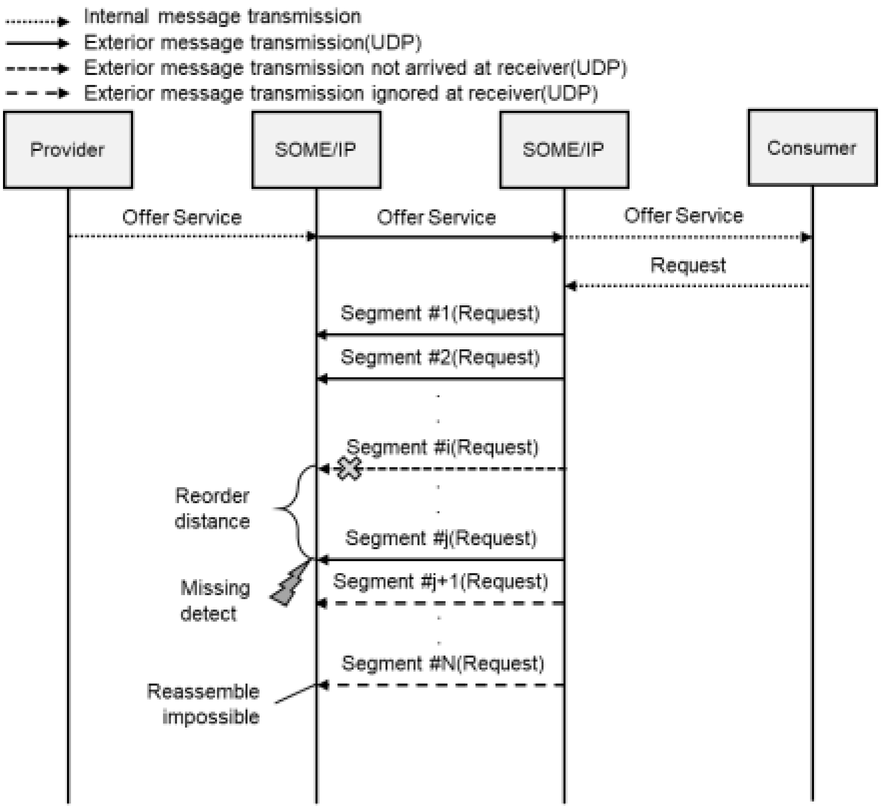

Figure 15. Problem case of SOME/IP-TP message transmission, Missing: Request case

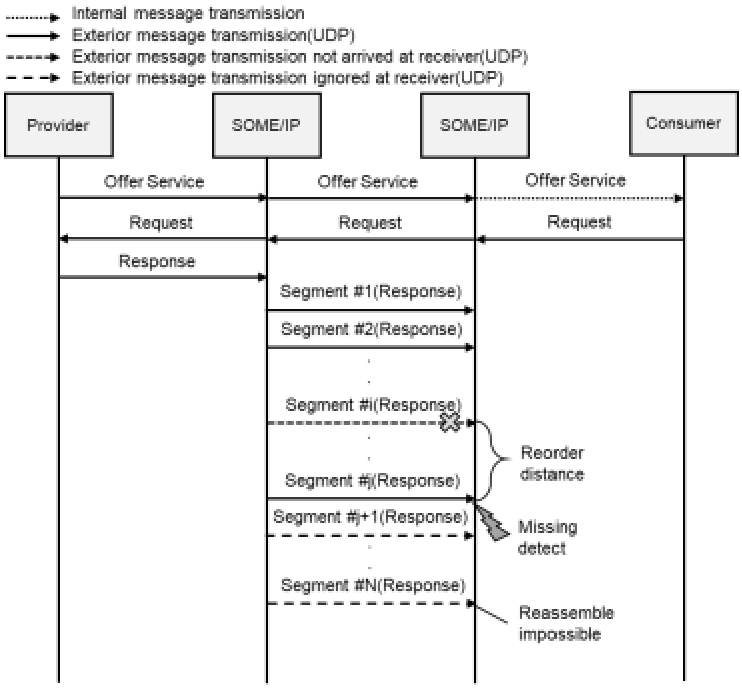

Figure 16. Problem case of SOME/IP-TP message transmission, Missing: Response case

As shown in Fig. 14, 15 and 16, when Missing occurs, a problem happens the Provider continuously sends segments that will be ignored by Consumer(segments #i~#N in Fig. 14, 15 and 16) These segments can't be reassembled into the original message. Therefore, transmission of these segments means the wasted data transmission. The wasted data transmission is a problem that makes the unnecessary load on the Provider and Consumer. Also, this is a problem that unnecessarily occupies bandwidth on the network.

5. Conclusions

In this paper, the problem when using SOME/IP-TP for handling big data in vehicles was analyzed. Problems were analyzed in three steps. First, we investigated the problem when using SOME/IP-TP caused by characteristics of UDP such as connectionless and unreliable. Second, we defined each exception that can be occurred when the data is transferring through SOME/IP-TP as Shutdown and Missing. Third, we analyzed problems that can be detected in defined exceptions. As a result, it was confirmed that wasted data transmission problems occurred when Missing of segmented data or Shutdown of the Response message receiver.

Previous studies about UDP supplemented the characteristics of UDP such as connectionless and unreliable[24, 25, 26]. However, these characteristics of UDP were not considered in SOME/IP-TP and there were no relevant studies.

In the latest version of the SOME/IP specification, the concept of 'Reorder distance' was deleted. In this case, the 'Missing', which is one of the two exceptions in this paper, may not be considered. However, the latest specifications cannot cover the problem that the order of message transmission can be changed frequently when using UDP. Therefore, there is a possibility that the specifications will be re-changed in the future.

Recently, SOME/IP is actively discussed and applied to actual vehicles, mainly in automotive OEMs and suppliers belonging to AUTOSAR[27, 28]. This paper has a meaningful significance that found a problem for big data transmission using SOME/IP in terms of traffic and stability that should be considered when applying to actual vehicles. Furthermore, it is expected that the results of this study can contribute to UDP-based big data handling research

References

[1] Bimbraw, K. (2015, July). Autonomous cars: Past, present and future a review of the developments in the last century, the present scenario and the expected future of autonomous vehicle technology. In 2015 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO) (Vol. 1, pp. 191-198). IEEE.

[2] Hussain, R., & Zeadally, S. (2018). Autonomous Cars: Research Results, Issues, and Future Challenges. IEEE Communications Surveys & Tutorials, 21(2), 1275-1313.

[3] Gupte, M. S., & Askhedkar, A. R. (2018, June). An Innovative Wireless Design for a Car Infotainment System. In 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS) (pp. 1751-1754). IEEE.

[4] Rao, Q., Grünler, C., Hammori, M., & Chakraborty, S. (2014, June). Design methods for augmented reality in-vehicle infotainment systems. In Proceedings of the 51st Annual Design Automation Conference (pp. 1-6). ACM.

[5] Bengler, K., Dietmayer, K., Farber, B., Maurer, M., Stiller, C., & Winner, H. (2014). Three decades of driver assistance systems: Review and future perspectives. IEEE Intelligent transportation systems magazine, 6(4), 6-22.

[6] Gerstmair, M., Melzer, A., Onic, A., & Huemer, M. (2019). On the Safe Road Toward Autonomous Driving: Phase noise monitoring in radar sensors for functional safety compliance. IEEE Signal Processing Magazine, 36(5), 60-70

[7] Bosch, R. (1991). CAN specification version 2.0. Rober Bousch GmbH, Postfach, 300240, 72.

[8] FlexRay Consortium. (2005). FlexRay communications system protocol specification. Version, 2(1), 198-207.

[9] LIN Consortium. (2003). LIN specification package, revision 2.0. Munich, Germany.

[10] Cooperation, M. O. S. T. (2004). MOST specification revision 2.3.

[11] Thomsen, T., & Drenkhan, G. (2008). Ethernet for AUTOSAR. EB Automotive Gmbh.

[12] Nichiţelea, T. C., & Unguritu, M. G. (2019, August). Automotive Ethernet Applications Using Scalable Service-Oriented Middleware over IP: Service Discovery. In 2019 24th International Conference on Methods and Models in Automation and Robotics (MMAR) (pp. 576-581). IEEE.

[13] Tuohy, S., Glavin, M., Hughes, C., Jones, E., Trivedi, M., & Kilmartin, L. (2014). Intra-vehicle networks: A review. IEEE Transactions on Intelligent Transportation Systems, 16(2), 534-545.

[14] Hank, P., Müller, S., Vermesan, O., & Van Den Keybus, J. (2013, March). Automotive ethernet: in-vehicle networking and smart mobility. In Proceedings of the Conference on Design, Automation and Test in Europe (pp. 1735-1739). EDA Consortium.

[15] Gopu, G. L., Kavitha, K. V., & Joy, J. (2016, March). Service oriented architecture based connectivity of automotive ecus. In 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT) (pp. 1-4). IEEE.

[16] AUTOSAR, Specification of SOME/IP, 2017, [online] Available: https://www.autosar.org/fileadmin/user_upload/standards/foundation/1-3/AUTOSAR_PRS_SOMEIPProtocol.pdf

[17] AUTOSAR, Example for a Serialization Protocol (SOME/IP), 2014, [online] Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/4-1/AUTOSAR_TR_SomeIpExample.pdf

[18] Coonjah, I., Catherine, P. C., & Soyjaudah, K. M. S. (2015, December). Experimental performance comparison between TCP vs UDP tunnel using OpenVPN. In 2015 International Conference on Computing, Communication and Security (ICCCS) (pp. 1-5). IEEE.

[19] Kuo, J. L., Shih, C. H., & Chen, Y. C. (2013, November). Performance analysis of real-time streaming under TCP and UDP in VANET via OMNET. In 2013 13th International Conference on ITS Telecommunications (ITST) (pp. 116-121). IEEE.

[20] Tomljenovic, D., Grbić, R., Lukač, Ž., & Basicevic, I. (2018, May). Performance Analysis of Protocol Stack for Inter-Processor Ethernet Communication in Automotive Industry. In 2018 Zooming Innovation in Consumer Technologies Conference (ZINC) (pp. 35-38). IEEE.

[21] Eggert, L. and G. Fairhurst, "Unicast UDP Usage Guidelines for Application Designers", RFC 5405, DOI 10.17487/RFC5405, November 2008, [online] Available: https://www.rfc-editor.org/info/rfc5405

[22] Fairhurst, G. and T. Jones, "Transport Features of the User Datagram Protocol (UDP) and Lightweight UDP (UDP-Lite)", RFC 8304, DOI 10.17487/RFC8304, February 2018, [online] Available: https://www.rfc-editor.org/info/rfc8304

[23] E. Farchi, Y. Krasny and Y. Nir, "Automatic simulation of network problems in UDP-based Java programs," 18th International Parallel and Distributed Processing Symposium, 2004. Proceedings., Santa Fe, NM, USA, 2004, pp. 267-.

[24] Y. Ge, B. Xie and Y. Song, "Design Based on Reliable UDP Multi-Window Transmission," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2018, pp. 113-116.

[25] Qifeng Sun and Hongqiang Li, "Research and application of a UDP-based reliable data transfer protocol in wireless data transmission," 2011 International Conference on Computer Science and Service System (CSSS), Nanjing, 2011, pp. 1514-1516.

[26] A. O. F. Atya and Jilong Kuang, "RUFC: A flexible framework for reliable UDP with flow control," 8th International Conference for Internet Technology and Secured Transactions (ICITST-2013), London, 2013, pp. 276-281.

[27] Staron, M., & Durisic, D. (2017). AUTOSAR Standard. In Automotive Software Architectures (pp. 81-116). Springer, Cham.

[28] AUTOSAR, Technical Overview v. 2.2.1., 2008, [online] Available: https://www.autosar.org/fileadmin/user_upload/standards/classic/3-0/AUTOSAR_TechnicalOverview.pdf