Implementation of Security Access Service using SHA-2 Algorithm

Shubhashree Holla
Dept of ECE
B.M.S. College of Engineering, Bengaluru

Akhila S., PhD
Professor
Dept of ECE
B.M.S. College of Engineering, Bengaluru

ABSTRACT
Advanced vehicles are not just a machine-like device; modern vehicles are continuously controlled, as well as monitored by using many digital computers, also called electronic control units (ECU). Advancement in modern automobiles mainly concentrates on the security and safety of these devices, thereby providing a safe driving environment to a driver. Radar is one of the important parts of the advanced driving assistance system, providing some important features like blind spot detection, cross traffic assistance, safe door opening etc. It is important that information coming from the Radar should be safe and it cannot be accessed by the third party. Aim of this paper is to implement the security access service to protect information coming from the Radar sensor. Application layer of the Autosar will allow the bidirectional communication between the client and the server. Implementing this service in application layer will reduce the time and it also provides increased level of security to the system. This paper comprise of implementation of this security access service using Unified diagnostic service (UDS) protocol.

Keywords
SHA-2, AES, Automotive, Security access service, AUTOSAR, UDS protocol

1. INTRODUCTION
The continuous improvement of living standards in constant changing world, buying and driving a car seems to have become necessary in people's life. Safety of a driver plays an important role while driving a car, because it includes different risk parameters in driver's life. Driver's security parameter primarily based on car's different products quality, during the manufacturing of a vehicle and it is mainly a manufacturer responsibility to provide a safe environment for driving.

Active safety along with passive safety plays major role in an automotive industry to provide safety to car drivers. Active safety systems always stay active while a person is driving the vehicle and these systems works continuously to keep driver from getting into an accident. Example for some active safety systems are Radar, Automatic break control system, Vision system etc. These active safety features are developed on the car's electronic control unit (ECU). International Organization for Standardization defines several Diagnostic services being used by the automotive industry. Security service can be implemented in the form of seed-key pair or request and response format in the application layer. Different diagnostic services are required to get the exact reason for the problem and also for troubleshooting the particular problem occurring in the vehicle while driving. These diagnostic services are implemented based on the unified diagnostic protocol standard. This automotive protocol helps the tester to effectively communicate with the cars electronic control unit to identify the fault and to get the appropriate solution to the fault.

This paper mainly concentrates on implementation of the security access service in application layer of the Autosar, taking the help of the Run time environment layer.

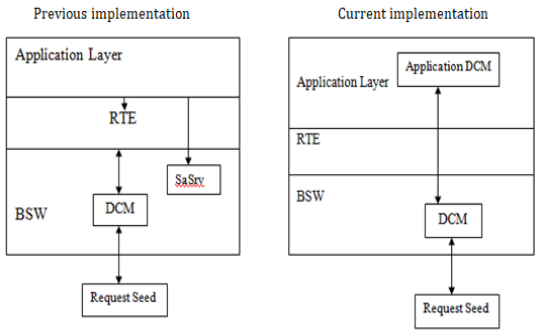

Fig.1: Autosar Structure

Fig.1 depicts the difference between previous implementation and current implementation of the security access service. Previously, to process the request message, diagnostic communication manager (DCM) present in basic software (BSW) layer communicates with run time environment layer to pass the message to SaSrv, it is a message processing segment in BSW layer. Even though DCM and SaSrv, both are present in the same layer of the Autosar i.e BSW layer, they cannot communicate with each other directly. This will increase the time delay. Hence security service is implemented in the DCM present in the application layer. Here basic software DCM directly communicates with the application DCM, to process the request message. Mainly this layer supports the SHA-2 algorithm. Diagnostic tester is referred as client and it is apparently connected to the server or it is also called as communicating medium. Diagnostic tester sends the request message to the cars electronic control unit though controller area network. Server responds back to the tester in the form of response frame according to the request made.

In UDS protocol, several vehicle diagnostic services are defined, which are communication control service, tester present service, read memory by address service , security access service, control DTC setting service etc. Each service has its own functionality, in order to check the cars status. All these diagnostic services are implemented in the form of software and this need to be flashed on the cars electronic control unit for the proper functionality of the vehicle. Automotive open system architecture (Autosar) is a standard platform for the implementation of different automotive

diagnostic service.

2. PREVIOUS WORK

In order to avoid different attacks which possibly happens on vehicles, some evaluation process is defined for security access service [3]. At first, regulator region was identified so that network ID can answers to each analytic transmission. At that point network ID identifies the data comparing to specific CAN ID. For recognized CANID, checked if there is any security access administration is exits or not and furthermore the relating security access level is stamped. After the completion of these steps they calculated the entire length of the seed value used for each detected security access service. A Controller Area Network (CAN) transport [1] is a standard bus network for car and plan of this transport standard is made so that, it permits microcontrollers and different gadgets to speak with one another's applications without the presence of host PC. Primary focus was to know the attacker's intension, in the event that he/she had the option to vindictively impart on the vehicle's inner matrix. In automotive field CAN bus security is the major challenge because all the ECU's are connected to the common CAN bus and these CAN packets are broadcast to all nodes by both physically and logically, it creates a several problem when there is hostile components are present in the network [4]. This revealed that the remarkable assaults don't need a total comprehension of a solitary segment of the vehicle. Actually, the scope of legitimate CAN bundles is little. Hence simple fuzzing of packets can make significant damage.

3. METHODOLOGY

3.1 Security access service (0x27)

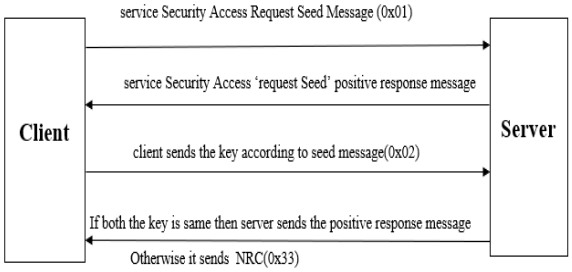

Fig.2: Communication between client and server for security access service (0x27)

The motivation behind security access diagnostic service is to give a way to get to information and indicative administrations, which have confined admittance for vehicle security, discharges, or wellbeing reasons. Fig.2 explains the message transmission between tester and user. SecurityAccess 'requestSeed' message is sent by the client to demand the tester to unlatch the security service. SecurityAccess 'requestSeed' positive response information is sent by the tester to client, this message includes the seed value. Then client will reply by giving back key value to the tester using SecurityAccess sendKey request information. The worker will analyze this key to one inside put away or determined. In the event that the two numbers coordinate, at that point the worker will empower the customer's admittance to explicit administrations or information and demonstrate that with the administration SecurityAccess sendKey positive reaction message. If both the keys are not similar to each other, then this shall be observed as false security access try. If an invalid key exists during the process, then entire process need to be start from the initial stage with requestSeed message.

3.2 SHA-2 or SHA-256

SHA-2 involves significant changes from its forerunner that is SHA-1. SHA-2 family consists of six hash functions with different hash values that are 224, 256, 384 or 512. SHA-2 or it is also called as SHA-256 is one of the strongest hash functions available.SHA-256 is a novel hash functions computed with 32-bit. SHA-256 and SHA-1 uses different additive constants and shift amounts, structures of these two algorithms are almost same. They differ only in number of rounds. SHA-224 is a truncated version of SHA-256, computed with different initial values. SHA-2 hash function is implemented for widely used security applications and protocols.

SHA-2 is used to compute the hash value of message M having length l bits, where length l is given as $0 <= l < 2^{64}$. This means message schedule of 64 thirty two bit words. It mainly consists of eight working variable and each of size 32 bits. Primarily these working variables are filled with constant eight initial hash values. The final result of SHA- 2 is a 256 bit message digests with given input as 512 bits. Here eight working variables are namely a, b, c, d, e, f, g, h and Initial hash values are labeled as $H_0, H_1 \ldots . H_7$. After processing each message block initial hash values are replaced by intermediate hash value and ends with the final hash value that is $H^{(N)}$.

3.3 Transport Layer Protocol

In Transport layer protocol message can be transmitted or received up to 4294967295 and state of the transmission or reception. Single frame and multiple frame transmission are defined under this protocol.

In Transport layer protocol message can be transmitted or received up to 4294967295 and state of the transmission or reception. Single frame and multiple frame transmission are defined under this protocol.

Single Frame: It is a transmission of message up to 6 or 7 bytes of data is performed based on transmission of unique N-PDUs. For example take unsegmented message with CAN_DL<= 8, is stored in the lower nibble of the PCI bytes. If CAN_DL>8, then message length is embedded in a second PCI byte.

First Frame: Transmission of longer message is performed by segmenting the message and transmitting the segmented message in N-PDUs. It consists of first set of data frames. Message length is transmitted in First Frame. By observing Fig.3, if message length is <= 4095 then this message length is stored in lower nibble of first PCI byte and second PCI byte. If message length is > 4095, then lower nibble of first PCI byte consists of 0000_2 and second PCI is filled with zeros and message length is present in a remaining four PCI bytes.

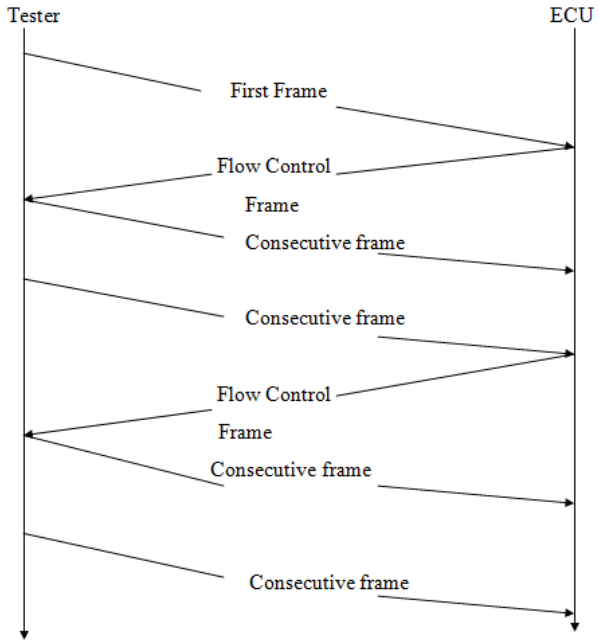

Fig.3: Flow Control Mechanism

Consecutive Frame: PDUs other than first PDU is called consecutive frames. While receiving the consecutive frame, the receiving network layer information shall collects the receive message bytes until the entire information is received. After collecting the last frame of the given message without any error, it can pass the received assembled message to the close to upper protocol layer.

Flow Control: Flow control is to regulate the rate at which consecutive frame N-PDUs are sent to the receiver.

3.3 GetSeed and CompareKey function

Mainly two functions are there in security access implementation, which are GetSeed and CompareKey. These Two functions are generated by using Davinci Configurator. It is a central tool for validating, configuring and generating the basic software and runtime environment of an AUTOSAR ECU.

GetSeed function is a request from the diagnostic communication manager to the application to provide a security level specific seed. This function includes operation status, seed data and negative response code. This function primarily fills the 15 initial vector buffers with zeros. In the next step 30 session key buffers are initialized with zeros. Next step is to reading the customer serial number also called ECU serial number from the PPAR Flash. PPAR Flash is a specific function where all the ECU serial numbers are stored. Then seed value is calculated using random value generator. These random values are generated with the timer, hence for every time this value changes. Then it checks if generated random value is equal to 0 or 0Xffffffff, then it returns the default seed value. In other case it returns the generated random value. Then calculating the new PUN values and storing it in a pun buffer. It is very much similar to the random value. Later these PUN values are used to fill the seed buffer. SHA-256 is used to generate the seed value using previously defined random values. AES encryption is performed to encrypt the data. After all the process whatever value present in the seed buffer is sent to the seed, and it is the final seed value received by the specific client.

Compare key function is a request from the DCM to the application to verify the requested security access level specific key. This function includes operation status, key data and negative response code. This function primarily fills the 15 initial vector buffers with zeros. Then seed buffer is filled with the values present in the Key buffer. Then AES decryption takes place. It is the reverse of AES encryption. At first mixing of columns will takes place then shifting rows and sub-byte operation will takes place. Finally the original plain text was obtained. After getting the key value from client, now server verifies this key with internally generated key value. If it matches then server will allow the client to access data. If it not matches then it will send the negative response message.

4. DETAILED EVALUATION OF SECURITY ACCESS SERVICE

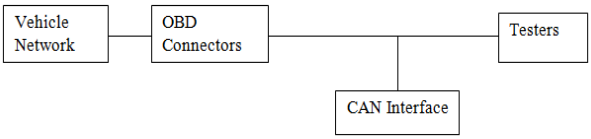

Fig.4: Physical setup to analyze the diagnostic communication

Fig.4 describes the physical setup to determine the diagnostic communications between vehicle ECU and tester, and also the message decoding capability according to the unified diagnostic service protocol. To record the correspondence between the individual vehicles ECU and analyzer, an extra customer was added to the diagnostics line, a transport investigation apparatus running on the connected PC. Accordingly, the current correspondence between various vehicles and the analyzer could be handily recorded.

In Fig.5 7d4 represents the transmit ID and 7d5 represents the receiver ID. 4th column in the message index represents the total length of the request message. Here total length of the request message is 8 bit and next column represents, out of 8 bit how many bits are used. Request message uses only 2 bits, which includes service ID (27) and sub service ID (01) and remaining bits are filled with 0xaa. Next tester will respond (7d5) by sending the first frame (10). It includes response ID (67) and subservice ID (01) and remaining 6 bits are filled with seed message. After receiving the first frame ECU sends the flow control frame (30) to ensure the consecutive message frame transmission from the tester. Then tester will send the reaming 26 byte seed message through consecutive frame (21,22,23,24). Fig.6 shows the 32 byte seed value transmitted from the tester with response ID 67 01.

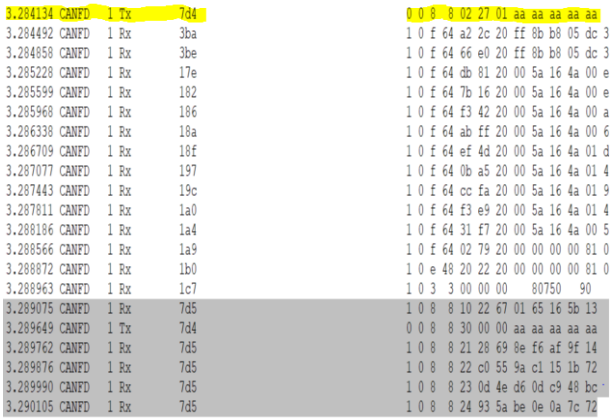

Fig.5: Request and Response message from the CanOe software

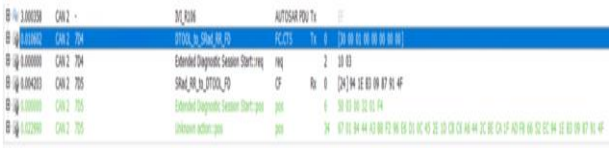

Fig.6: 32 byte seed value transmitted from the tester

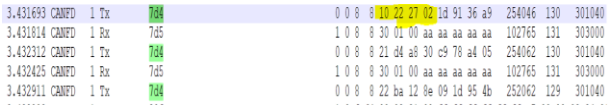

Fig.7: 32 bit key sent from the ECU

After receiving the seed value, vehicle ECU starts to calculate the key value. Then this key value is sent back to the tester (27 02) Fig.7. At the same time tester will also calculates the key value for specific seed. After receiving the 32 bit key from the ECU, tester will compares these two keys. If these two keys are similar then tester responds with a positive response (67 02) Fig.8. Which means security access is relaxed, now the particular ECU can share the radar sensor information. If both are keys are not matched then tester will send the negative response message.

Fig.8: Positive response message from the tester

4.1 Negative Response (0x7f)

This section describes some negative responses. Security Service supports some negative responses, for example if sub function ID is other than 01 or 02 then server shows 0x12 NRC. Similarly some more NRCs are supported in 0x27 services.

NRC 13:

NRC 13 exists only when the message length is not correct or format of message is not correct. By observing Fig.9 one can say that length of the message is 03 which is not correct, every message length should be 8 but here it is 3, so there exist NRC 13.

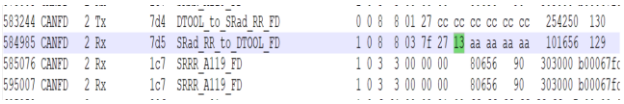

Fig.9: NRC 13

NRC 24:

This NRC exists when send key sub function arrives before the seed request. Seed request should come first, otherwise it is considered as NRC 24, shown in Fig.10.

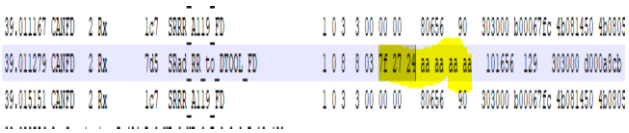

Fig.10: NRC 24

5. CONCLUSIONS

Establishing a software tool provides the security access service to the special ECU of the vehicle and to analyze the communication between automobiles and diagnostic testers. In order to provide trust in mobility, quality and safety are the important things in automotive industry. These features can be achieved by using AUTOSAR diagnostic service. This paper concentrates on the implementation of security access service to protect the information coming from the Radar sensor of the modern vehicles. SaSrv supports only SHA-1 algorithm and using RTE layer will increase the time delay. To overcome this drawback, instead of implementing this service in BSW layer, this can be implemented at the application layer of the Autosar. Application layer supports the SHA-2 algorithm. In SHA-2 32 byte of seed value can be exchanged between client and server. Basic software DCM directly communicates with the application DCM. Server shares the 32 byte seed value to the client, based on this seed value client will calculate the 16 bit key, and then shares key value to the server. Later sever contrast key along with generated key at the server. If both the key matches then server will provide the access to the client. AES-128 encryption is used to encrypt the information. This complete software can be tested using the canoe software. Davinci tool helps to generate the functions required for the establishment of the communication.

These days in the cutting edge vehicle one can make profit by the on board diagnostics. In future security measures can be expanded while driving. Model: connecting the GPS subtleties to the analytic administrations at whatever point an issue or mishap has happened. Likewise of late remote strategies would be executed by the fix shops to get the information, where demonstrative administrations will assume a significant job. This symptomatic information is acceptable abundance of information which is utilized for investigation of a vehicle condition and expansion of new highlights. In future, greater usefulness can be expected which exploits indicative administrations directly from the dashboard. Subsequently later on, a period of self driving self-sufficient vehicles with self revising capacity of mistakes can be normal.

6. ACKNOWLEDGEMENT

The authors would like to acknowledge Mr. Mohanbabu Penagalapati and Ms. Rashmi Kori of Veoneer India Pvt Ltd, Bengaluru for giving the opportunity to work in the field of automotive electronics.

7. REFERENCES

[1] ISO 15765-3:2004, “Road vehicles- Diagnostic on controller area network (CAN)”

[2] ISO 14229-1:2013(E), “Road vehicles -Unified diagnostic services (UDS)-Part 1: Specification and requirements”.

[3] Ryo Kurachi, Hiroaki Takada, Kentaro Takei, TakaakiIinuma, Yuki Satoh, Manabu Nakano, Hideki Matsushima, Jun Anzai, Toshihisa Nakano, “Evaluation of Security Access Service in Automotive Diagnostic Communication”, 2019 IEEE 89th Vehicular Technology Conference, Kuala Lumpur, Malaysia.

[4] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, and Tadayoshi Kohno, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, HovavShacham, and Stefan Savage, “Experimental Security Analysis of a Modern Automobile”, 2010 IEEE Symposium on Security and

Privacy.

[5] Martin Ring,TobiasRensen and Reiner Kriesten, "Evaluation of Vehicle Diagnostics Security - Implementation of a Reproducible Security Access", SECURWARE Portugal, 2014.

[6] Parag Kharche, Meera Murali and Geetanjali Khot, "UDS Implemetation for ECU I/O Testing", IEEE 2018 3rd International Conference on Intelligent Transportation Engineering.

[7] M.Salcianu and C.Fosalau, "A new CAN diagnostic fault simulator based on UDS protocol", 2012 International conference and Exposition on electrical and power engineering, Iasi, pp, 820-824, 2012.

[8] Lupei and L. Stanciu, "Application for UDS automated test generation", 2016 IEEE 1^{1th} International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, pp. 85-88, 2016.

[9] Secure Hash Standard (SHS), Information Technology Laboratory National Institute of Standards and Technology, August 20.

[10] ISO 15765-3:2016, "Diagnostic communication over controller area network (DoCAN) - Part 2: Transport protocol and network layer services."

[11] AUTOSAR, "Specification of Diagnostic Communication Manager", Document ID-18, 2017.