

Security, Vulnerability and Protection of Vehicular On-board Diagnostics

Aastha Yadav¹, Gaurav Bose², Radhika Bhange³, Karan Kapoor⁴,
N.Ch.S.N Iyengar⁵, Ronnie D. Caytiles⁶

¹⁻⁵ VIT University, Vellore-632014, Tamilnadu, India, ⁶ Hannam University, Korea
{¹aasthay1705, ²gb23295, ³radhika.bhange, ⁴karankapoor091}@gmail.com,
⁵nchsniyr@vit.ac.in

Abstract

Recent studies have shown that ‘hacktivists’ can mount serious attacks on automobiles. The automotive On-Board Diagnostic (OBD) interface enables an efficient way to access information of the in-vehicle electronic system and leaves way for unauthorized access by an intruder. We discover that remote exploitation is feasible via a broad range of attack points such as mechanic tools, CD players, Bluetooth and Tire Pressure Monitoring System. Wireless communications channels allow long distance vehicle control. Finally, the paper discusses the drawbacks of Seed-Key Mechanism to authenticate and provide an extra layer of authentication to help build a safer automotive ecosystem.

Keywords: On-Board Diagnostic (OBD), Tire Pressure Monitoring System, Seed-Key Mechanism

1. Introduction

While most modern vehicles can be hacked, access to a car's internal network can circumvent all computer control systems, including critical safety elements such as the brakes and engine. It is possible for an attacker to take control of a car and its various features and functions without ever having hands-on access to the car. A car is a mini network. Currently there is no security implemented. Hackers can disable immobilizers and drive off without a key in models from Volvo, VW, Audi and Fiat. Projects like SeVecom are developed to provide secure vehicular communication systems to make driving safer [17].

The real motive is to reduce the cost of research in the field of car hacking by standardizing and simplifying investigation techniques to the point where interested researchers won't need an actual car. Our digital lives aren't private and neither are anything we click, swipe and search in our cars. Automakers in partnership with big data firms like SAP, are trying to turn vehicles into one-stop shopping centers. Future version of cars like Audi MMI, might be able to pay for gas directly from the car. At that point, hackers worldwide will have every incentive to scan the onboard hard drive and anything else connected to the car, looking for credit cards, addresses, social security numbers, passwords and other personal information.

2. Motivation

Automobile hacking has become one of the major concerns for security today. The attacker can take control of your car and its various features and functions without having hands-on access to the car. Everyday a new method of hacking is developed, which challenges the existing security solutions.

Different solutions for security already exist. Most of the automobile companies try to implement these in their new models to add up on the security level of the car. These solutions are clearly very costly and sophisticated but they are still not very efficient to completely block the hacker.

Secondly, whenever someone unauthorized is trying to gain access to the car, a key is entered to check the authentication. This security system fails on many levels. Hence if there is an additional level to the existing security system, it will not only add extra security but also will notify the owner instantly about any misuse or unauthorized entry. The idea is to have a totally secure system that will make it almost impossible for the hackers to enter into the system.

3. Background Research

3.1. Digital Embedded Systems

Currently four out of ten car thefts in major cities like London involve some form of car hacking. In light of police reports being unable to explain how some cars are stolen, researchers Roel Verdult, Flavio Garcia and Baris Ege began to investigate the nature of vehicle immobilizer's devices that prevent the engine from starting without the correct key. Immobilizers used in 100 different models from the likes of Volvo, VW, Audi and Fiat, especially models that come with a starter button instead of a key were found vulnerable to hacking by thieves with access to a computer. The researchers were banned from publishing the report for two years by car manufacturers due to its sensitive nature [1]. Today's automobiles contain a number of different electronic components networked together that as a whole are responsible for monitoring and controlling the state of the vehicle. Each component, from the Anti-Lock Brake module to the Instrument Cluster to the Telematics module can communicate with neighboring components. Modern automobiles contain more than 50 electronic control units (ECUs) networked together. The overall safety of the vehicle relies on near real time communication between these various ECUs. While communicating with each other, ECUs are responsible for predicting crashes, detecting skids, performing anti-lock braking, *etc* [2].

3.2. Automobile Network

ECUs don't exist in isolation; they are coupled to one another via Controller Area Network or CAN, a requirement in U.S. cars since 2008. Automobile CAN networks were never set up for data security and still aren't. CAN is a series of data pipelines that connect various computers throughout the vehicle. It regulates commands sent across an entire network, prioritizes them and rejects faulty signals, but these networks were never designed to encrypt messages or shield the network from malicious attacks. The CAN bus was developed in 1983 for efficient and fast automotive communication and is standardized in ISO 11898 [15]. Automakers separate these CAN networks by what makes sense from a production standpoint, not because they have an innate desire to protect components like the steering and throttle from intruders [3]. Strategy Analytics estimates that by 2007, approximately 55 percent of all new cars will have a telematics capable terminal, as compared to approximately 7.5 percent in 2000 [18].

4. Points of Entry

4.1. Physical Access

The hackers might need physical access to the cars, sometimes to the point of ripping apart the dashboards, to initiate the hackers wired laptop connection to ultimately shut off

the vehicle. [8] The attackers can connect their laptops through wiring or cable. They can get access through gadgets plugged directly into cars most sensitive guts. [9]

4.1.1. Door Locks and Key Fobs: An attacker could emulate the presence of access code used by these two systems. In this way he could control locks and start or stop the car engine [6].

4.2. Wireless

4.2.1. Bluetooth System: It is a popular feature that allows an owner to pair their phone with their car, but it also provides a wireless point of entry for hackers. Hackers can compromise an owner's phone by trying to entice them to visit a malicious website. Or they can capture a phone's MAC (media access control) address when someone starts their car while the phone is paired to it. Hackers can also capture the MAC address by sniffing the Bluetooth traffic generated when any wireless paired device in a car has its Bluetooth unit enabled. [10] A 3G capable, in-car navigation is an easy target because these systems communicate over long-range cellular networks and are ideal for remote exploitation.

4.2.2. CD Player: This can be corrupted with malicious code and used as a way to get in. Hackers could use social media to entice drivers to download a song and play it in their car. The car's media player would then display a cryptic message, and if the driver did not press the right button, it would re-flash the unit and load it with corrupted software.

4.2.3. Tire Pressure Monitoring System (TPMS): It is a government mandated display that shows the air pressure of each tire on a screen in the instrument panel. That information is wirelessly transmitted from each wheel to a computer inside the car. The TPMS relies on radio frequency identification because they're wireless [10]. In the case of TPMS hack, wireless tire sensors communicate once every 60 to 90 seconds infrequently [11].

4.2.4. Unauthorized Applications: Cars are equipped with on board computers that can execute or download applications. For cars, these applications can be provided by malicious and unauthorized third-parties; This is generally handled through application stores or dealer customized interface [6]. There is a method for side loading the applications for testing which can be used to execute malicious codes to further unlock the system [19]

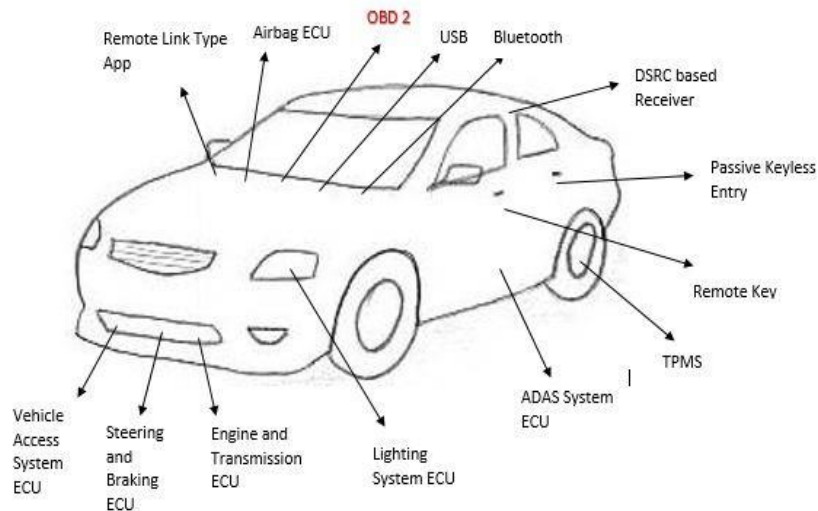


Figure 1. Entry Points of Attack

5. On-Board Diagnostic (OBD)-II Port

Modern vehicles have become complex information technology (IT) systems consisting of multiple interconnected Electronic Control Units (ECUs) which are responsible for safe and correct functionality of the car. In order to provide comprehensive, easy-to-use self diagnostic and reporting functionality for in vehicle ECUs, a standardized On-Board Diagnostic (OBD) interface was developed in the 1990s and today OBD is deployed worldwide and legally mandatory in the US and Europe. OBD-II is a sort of computer which monitors emissions, mileage, speed, and other useful data. OBD-II is connected to the Check Engine light, which illuminates when the system detects a problem. On one hand, OBD enables digital access to public data, for instance to emission control and error codes and on the other hand, OBD enables access also to “hidden” manufacturer-specific ECU settings, for instance to theft protection or engine control [4]. The problem for hackers is that the OBD port is typically inside the car, and this direct access requirement makes it an unlikely target for a completely random, unknown attacker. [7] Even though the OBD standard foresees some basic mechanisms that enforce access control using four different security access levels, their practical realization is rather weak. Based on static, proprietary, or simply incorrect implemented algorithms, they can often be circumvented with cheap tools and only little publicly available knowledge [2]. Diagnostic Trouble Codes (DTC) is stored in the system. The codes are not necessarily the same across all vehicles, and it is not uncommon for foreign manufacturers to use manufacturer specific codes. Aside from that, however, a mechanic (or anyone with an OBD-II scan tool) can connect to the port, read the DTC, and identify the problem (or problems) with the vehicle.



Figure 2. OBD2 Port as Data Link Connector

The standard on-board diagnostic pin-out diagram is given in figure 2. If the vehicle's diagnostic port does have pins with access to the target bus, then one can take a scan tool and swap wires from the standard pins to the target bus pins to get access to the network. Otherwise this can be done by actually splicing into a wire harness somewhere in the vehicle. An OBD-II extension cable can be used to hack the vehicle end to give you raw wires to play with. The messages can now be sent into the bus to control components as the port is interconnected to various ECUs.

6. OBD-II Port Security Threats

A moderately priced Sedan comprises of less than 30 ECUs which includes both critical drive train components as well as less critical components such as windshield wipers, door locks and entertainment functions. It has been found out that a set of messages and signals that could be sent on car's CAN bus (via OBD-II) to control key components (*e.g.* lights, locks, brakes, and engine) as well as injecting code into key ECUs to insert persistent capabilities and to bridge across multiple CAN buses [13]. Due to commonly available tools and information, automotive manufacturers face an increasing amount of ECU manipulations that might affect vehicle safety, legal applications (*e.g.*, exhaust gas treatment) or undermine aftermarket business models. Thus, not only ambitious vehicle owner can misuse diagnostic services for illegal feature activation, ECU parameter manipulations or mileage reset. Also criminal organizations aim to retrieve information about the intellectual property of OEMs and suppliers for creating counterfeit components or about driver sensitive data such as driving behavior [4]. Furthermore, critical data are stored in the ECUs such as crash data, data for insurances, or warranty indicators. Such data is also very attractive for malicious manipulations. For example, data such as vehicle speed, seat belt status, brake pedal position *etc.* are typically recorded in the seconds before a crash. A driver who has been involved in an accident could be motivated to change the recorded data to indicate that the brakes were applied when they really were not [5].

7. Security Solution

7.1. Seed Key Algorithm

The “Seed-and-Key” algorithm applies a secret key value to calculate the response key from the seed.

Only tester in possession of the correct secret value of the ECU can answer the seed correctly and gain access to the diagnostic service. The problem is that the same secret key value is often used for a whole production line of a vehicle, *i.e.* the same ECU in all cars is using the same secret key value and sometimes this secret key value is shared by many ECUs. If an attacker can get hold of this globally used secret key value, he is able to access the diagnostic service of the whole production line. There are various ways to get hold of the secret key value, for instance from the garage or by extracting the secret key value from a tester. “Seed-and-Key” algorithms used by the manufacturers are proprietary and are considered as confidential. Another problem is that the secret key material is often stored in unprotected memory within the tester or within the ECU. An attacker who has access to a tester device or ECU can read out the secret material from this device, for instance, by gaining access to a tester over a debug interface. The possession of the secret key material now enables the attacker to execute the authentication in the protocol [4].

7.2. Working Mechanism

The seed key protocol is rife with holes and needs to be fixed by additional security. These are additives that address the major concerns and drawbacks of the simple seed key protocol as the communications after a successful authentication were still insecure.

Brute force attack requires 7.5 days to crack a 16-bit long seed and key values but the same can be cracked in less than 10 minutes [4][5]. This is done by using a device that predicts the polynomial used for seed key generation and then gets access to the global seeds and thereby the key. A strong tester authentication can be done in various ways [14].

The two solutions as proposed in the following section, work synchronously with the given seed-key protocol and makes seed and key values difficult to crack because the client is kept informed at all times.

7.2.1. Two-Way Authentication Method: The two-way authentication method adds an additional feature to the seed key protocol by means of a product that will allow the client to control a person from accessing the OBD-II port. The following architecture as described in the diagram 1 and Figure 3 describes the flow of the method used.

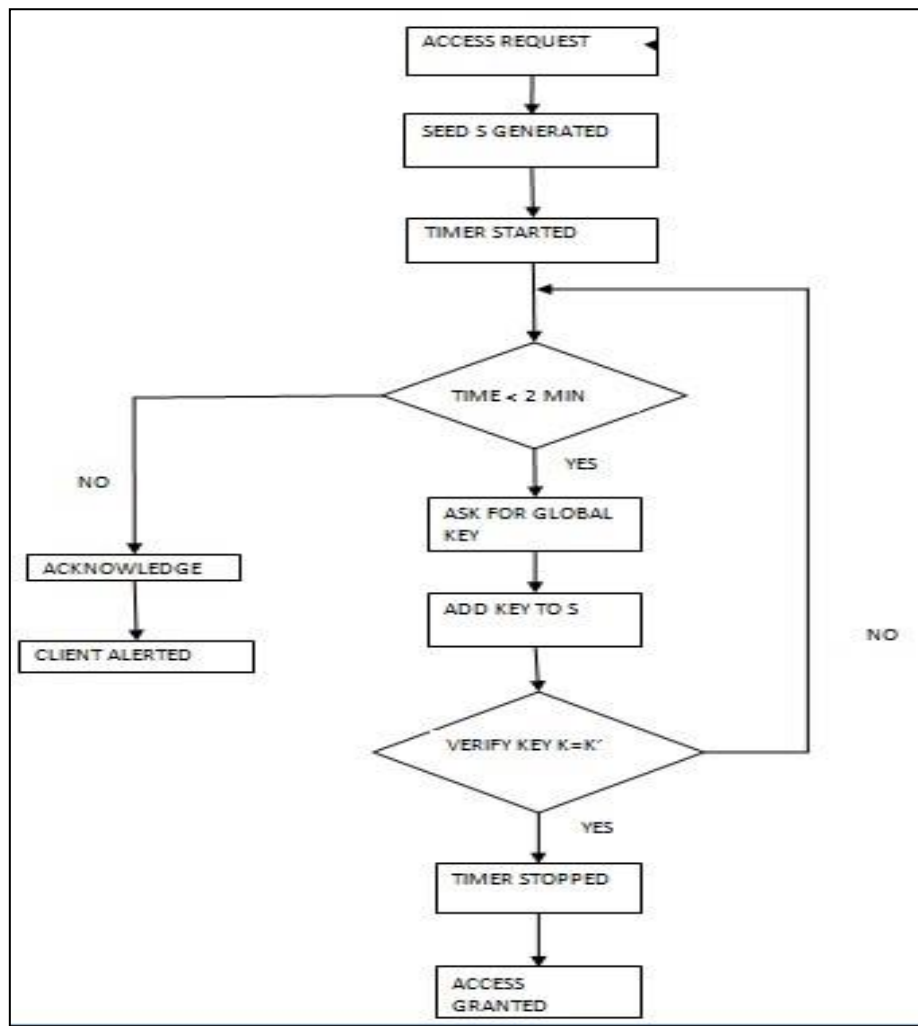


Diagram 1. Describing Two-Way Authentication

Here, the security access request is received and then a seed S is sent back to the tester. Once the tester receives the seed it communicates with the client device, it can be a pager, a cellular device or any device that will allow the client to receive and send an acknowledgement thereby completing the 2-way authentication process.

If the client decides to deny access, then the seed is dropped instantaneously and the response is not processed further. This process goes on till the client acknowledges the request and allows the global seed to be added to seed S. Upon acknowledgement, the authentication is complete and the user is given access.

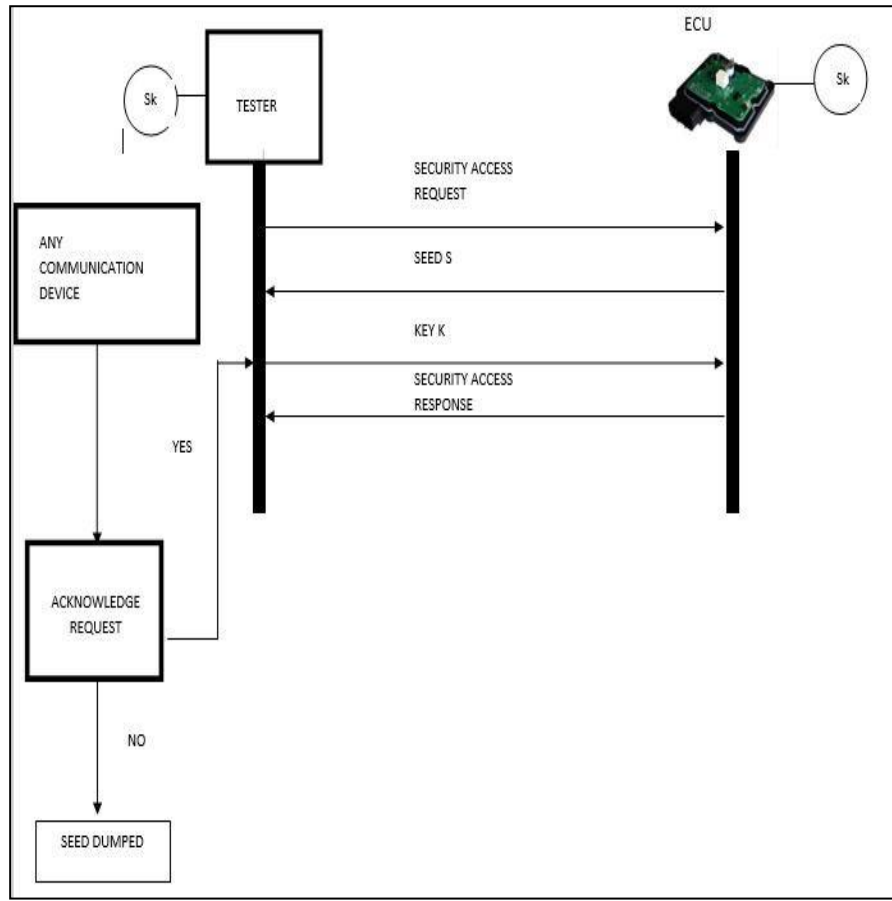


Figure 3. Two-Way Authentication and Seed-Key Mechanism

The above process adds a layer of safety as a result of keeping the client informed at every stage. The security method binds the advantages of seed key protocol and the proposed two-way authentication method to provide a safer method.

7.2.2. Timer Method : The timer method focuses and exploits the time brute force methods and other algorithms take to crack a 16-bit long seed key. The method is described in the flow chart (diagram 2).

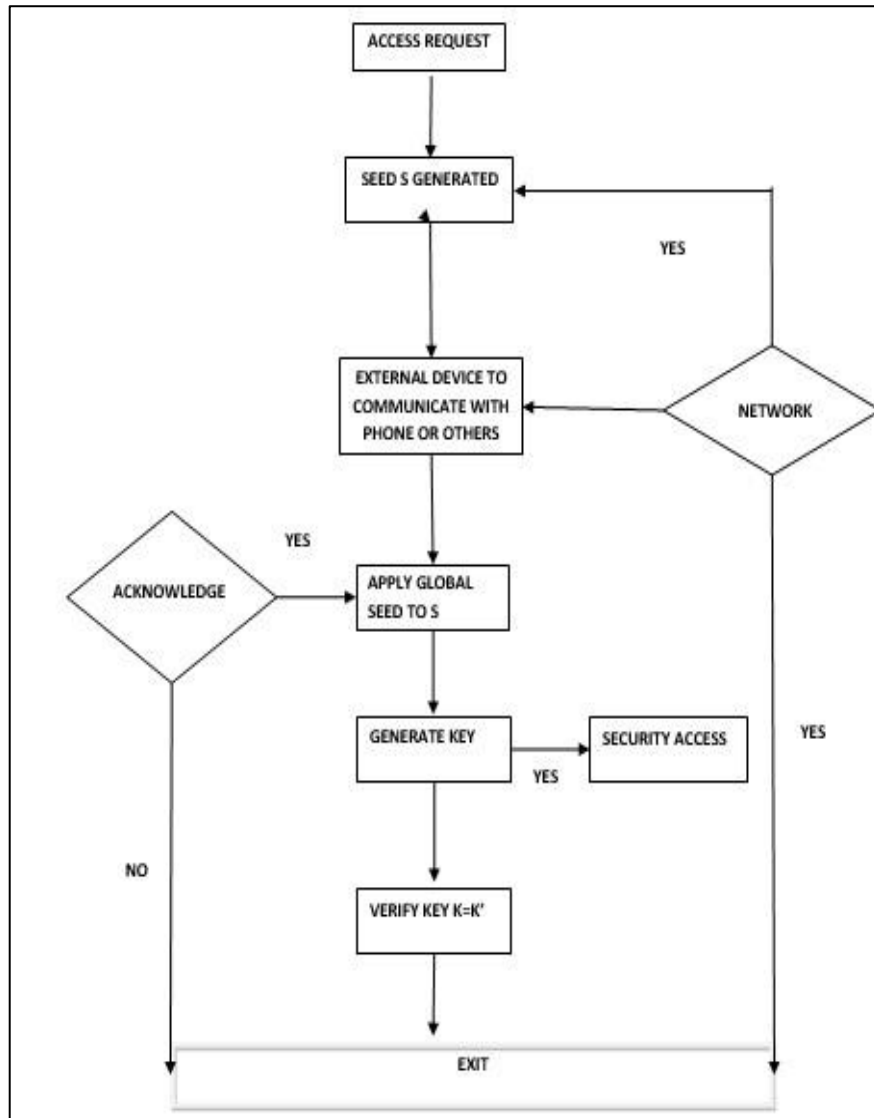


Diagram 2. Flow Chart describing Timer Method

Timer method gives more autonomy to the client as it directly gives the global seed to the client. This information is crucial as it is updated on a daily basis to implement entropy and randomness thereby adding a sense of difficulty. As seen above the security access request is received by the ECU and as soon as the seed S is sent back to the tester, the timer gets started.

The trial and error version is tested. The timer is set for a minute and if the client is assumed to take less than a minute to enter the global seed key, the seed key is added to the global seed S and the generated key is verified. When a malicious hacker uses a device to crack the algorithm, it takes more than a minute or the timer limit set by the user. OBD-II port is secured because as soon as the timer runs out, a message or a notification alert is sent to the client informing about the malicious activity as explained in Figure 4.

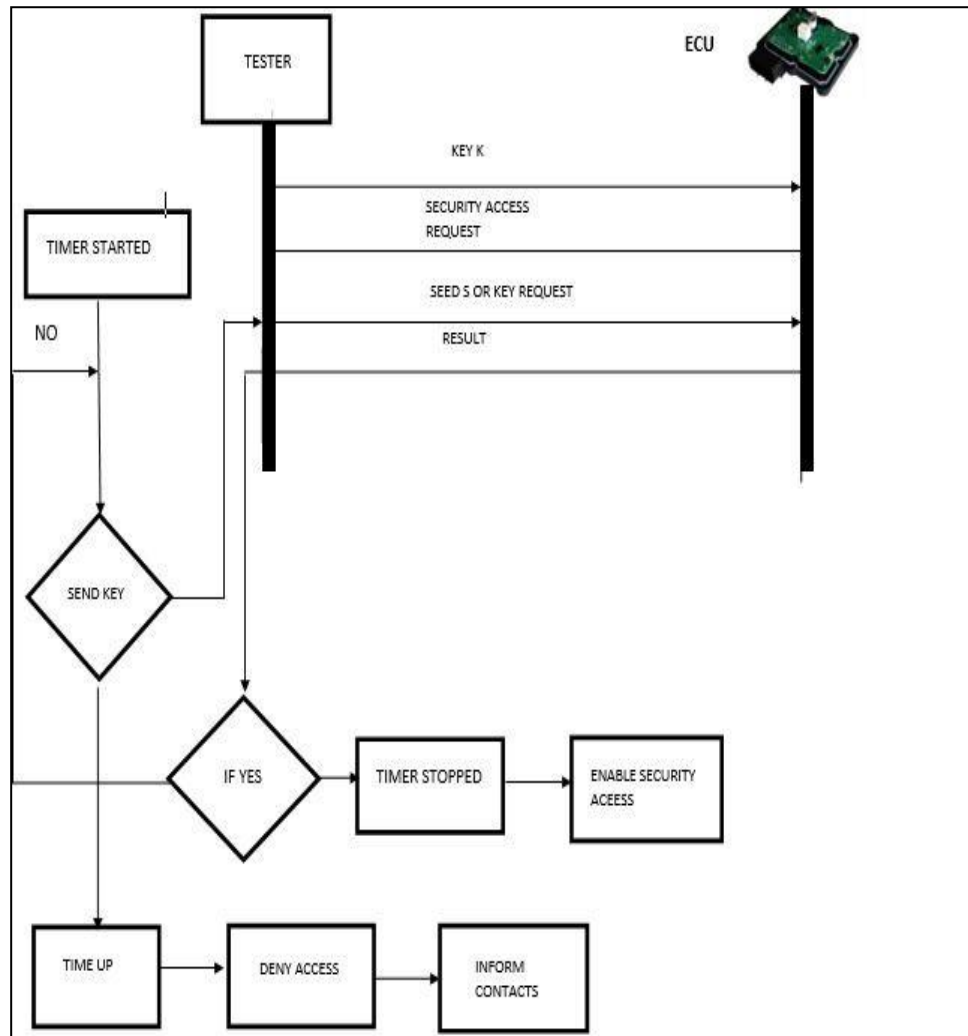
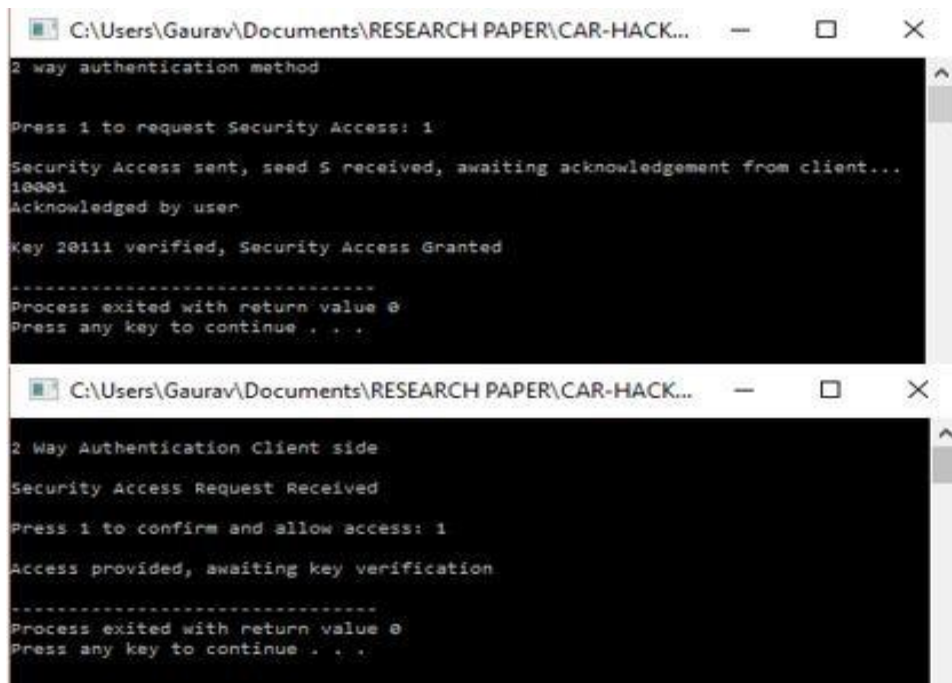


Figure 4. Timer Implementation in Seed-Key Mechanism

7.3. Implementation

The following section explains the code implementations of the above proposed methods to authenticate. The client server programs were run in the Cygwin environment on Windows and compiled by the gcc compiler.

7.3.1. Two-way Authentication Method : An execution requests for an access. Security access is provided each time an acknowledgement is sent by the client (through mobile phone, Bluetooth or any external device), thereby verifying the seed-key value to provide the access to the port as requested (Figure 5).

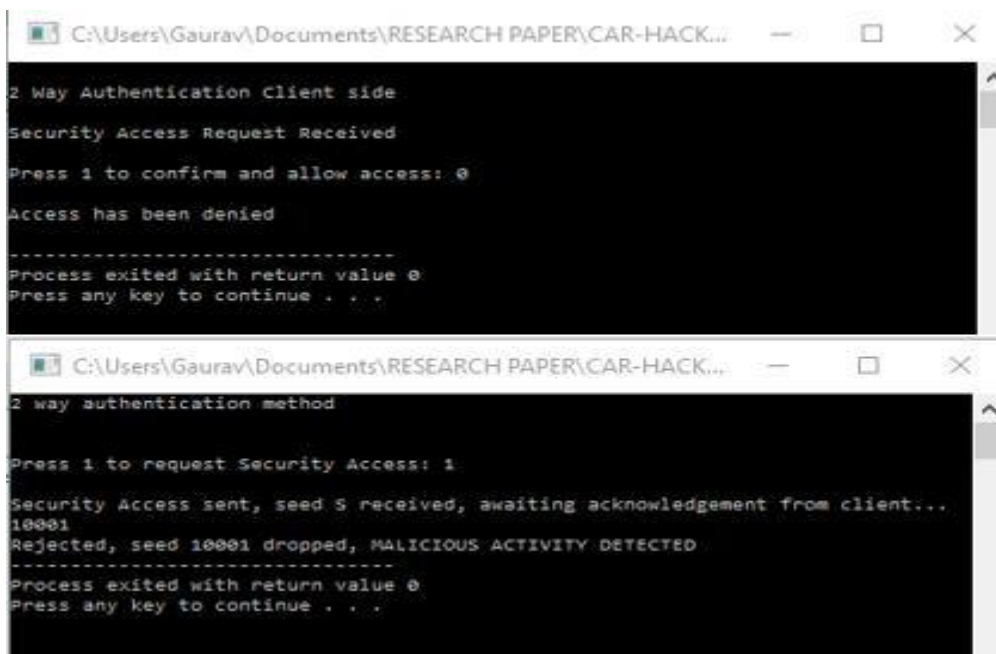


```
C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
2 way authentication method
Press 1 to request Security Access: 1
Security Access sent, seed 5 received, awaiting acknowledgement from client...
10001
Acknowledged by user
Key 20111 verified, Security Access Granted
-----
Process exited with return value 0
Press any key to continue . . .

C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
2 Way Authentication Client side
Security Access Request Received
Press 1 to confirm and allow access: 1
Access provided, awaiting key verification
-----
Process exited with return value 0
Press any key to continue . . .
```

Figure 5. An Access Request Verified

The two-way authentication is not complete until the client sends an acknowledgement to the request it receives, hence the access to the port is denied for the attacker (Figure 6).



```
C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
2 Way Authentication Client side
Security Access Request Received
Press 1 to confirm and allow access: 0
Access has been denied
-----
Process exited with return value 0
Press any key to continue . . .

C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
2 way authentication method
Press 1 to request Security Access: 1
Security Access sent, seed 5 received, awaiting acknowledgement from client...
10001
Rejected, seed 10001 dropped, MALICIOUS ACTIVITY DETECTED
-----
Process exited with return value 0
Press any key to continue . . .
```

Figure 6. Acknowledgement not received, Access Denied

However, when the client sends an acknowledgement for the access request, according to the seed-key mechanism, it now checks for the value of seed-key which if not verified leads to denied access to the user as the authentication is incomplete (Figure 7).

```

C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
2 way authentication method

Press 1 to request Security Access: 1

Security Access sent, seed S received, awaiting acknowledgement from client...
20001
Acknowledged by user

Key 38111 does not match, Security Access Denied

-----
Process exited with return value 0
Press any key to continue . . .

C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
2 Way Authentication Client side:
Security Access Request Received

Press 1 to confirm and allow access: 1

Access provided, awaiting key verification

-----
Process exited with return value 0
Press any key to continue . . .

```

Figure 7. Acknowledgment Received but Seed-Key Value is not Verified

Seed S sent by ECU is 20001

Table 1. Available Accepted Keys in the ECU

20111	20112	20113	20114	10011	10012	10013
-------	-------	-------	-------	-------	-------	-------

Table 1 values are to be verified after adding global key of 10110 to Seed S. Acknowledgement sent by client is 1, but authentication is incomplete because the generated key does not match available ECU values.

7.3.2. Timer Method: In the Timer method for authentication, if the secret key value matches one of the global key values used by the ECUs, before the timer goes out as described in the Figure 8, the authentication is complete and access is granted. However, if the secret key value does not match the global seed-key values on repeated attempts, and the timer goes out, access is denied for the intruder providing additional safety over the seed-key protocol (Figure 9).

```

C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
Timer authentication method

Press 1 to request Security Access: 1

Seed 10001 generated

Enter a key to be added to seed S
10111

Access Granted
Time Elapsed: 0:0:0:1

-----
Process exited with return value 0
Press any key to continue . . .

```

Figure 8. Seed-Key Value Verified before the Timer Goes Out

```

C:\Users\Gaurav\Documents\RESEARCH PAPER\CAR-HACK...
Timer authentication method

Press 1 to request Security Access: 1

Seed 10001 generated

Enter a key to be added to seed S
20000

Enter a key to be added to seed S
30000

Enter a key to be added to seed S
10000

Enter a key to be added to seed S
40000

Time is out 0:1:0:6 MALICIOUS ACTIVITY DETECTED. CLIENT INFORMED
-----
Process exited with return value 0
Press any key to continue . . .

```

Figure 9. Seed-Key Value is not Verified and the Timer Goes Out

Seed S sent by ECU is 10001

Table 2. Available Accepted Keys in the ECU

20111	20112	20113	20114	10011	10012	10013
-------	-------	-------	-------	-------	-------	-------

Table 2 values are to be verified after adding the global key X to seed S.

X is given by the client. Timer of 1 minute is set.

Case 1:

Seed S: 10001

Client enters: 10111

Generated Key (sum) = 20112

Verified in the array hence access granted.

Case 2:

Seed S 10001

Table 3. Client Keys and the Generated Keys

Sr.No	Entered Values	Key generated
1.	20000	30001
2.	30000	40001
3.	10000	20001
4.	40000	50001

None of the seed key values entered as described in table 3 were verified as the timer exceeds one minute, hence access is denied. Malicious activity is reported. Hence this method verifies the Seed S by identifying the known acceptable keys in the ECU along with the timer condition, providing an extra level of security.

8. Secure Vehicle Ecosystem

Connected cars bring tremendous promise for automakers and customers alike. Connectivity also brings in new risks. Stolen cars and online videos are only the tip of the iceberg compared to what could come. Still, building security into cars end to end will take many years. The existing connectivity within the vehicle opened up by the introduction of telematics services, and services provided over the Internet such as infotainment. In addition, OBD II port can be accessed via Bluetooth or Wi-Fi dongles, like the ones that insurance companies hand out so that drivers can prove they are driving safely and get a reduction on their insurance premiums. All of a sudden, what used to be a closed system has become an open one, but security measures have not always caught up [20].

On-board systems were designed to be wired and peer-to-peer, and as such were reasonably secure – but now they have been opened up by the use of wireless dongles, which may not even include authentication mechanisms [22].

The truth is that only the Original Equipment Manufacturer (OEM) is in a position to address the overall security of the vehicle and ecosystem – for example, it would be possible for them to introduce a hardware security module to authenticate the driver to all the systems inside the vehicle (not something an individual supplier can do). OEMs need to start considering connected vehicles as part of a larger system that includes vehicles, networks, OEM IT, and all kinds of services from third party providers, and to take overall responsibility for the security of the entire ecosystem.

8.1. Connected vehicles and the Internet of Things (IoT)

The connected vehicle raises some of the same issues as the IoT generally, including the need for integrity and authentication. However, vehicles are atypical in terms of their high value and long lifecycle. In addition, a vehicle is harder to update as it may involve a workshop visit and the cost of failure in terms of both money and human life is potentially

much higher than average. Among automotive sector respondents, 65% agreed that security concerns will impact customers' purchase decisions for IoT products. Just 35% of automotive respondents rated the IoT products in their industry high on resilience to cyber-attacks [21]. To provide authenticity, authentication and authorization should be integral to both on-board and off-board systems. The challenge may look daunting but to a great extent it is about adopting established good practice from software development (including mobile and IoT) and applying it to the broader requirements of the automotive industry.

9. Conclusion

This contribution gave an overview of various security vulnerabilities and points of entry. It was demonstrated that insufficient protection of the OBD interface leads to a high security risk which will be a threat to the user and the manufacturers. Hence the paper focuses on providing efficient security solutions to minimize the risk of the attacks through the OBD interface.

An additional layer of security is added to the seed-key mechanism which works as an authentication over the port. This paper proposes two new ways of providing additional security *i.e.* the two-way authentication and the timer method. With this information, individual researchers, consumers and manufacturers can propose ways to make ECU's safer in the presence of a hostile CAN network as well as ways to detect and stop CAN bus attacks through the OBD-II port. This will lead to safer and resilient vehicles in the future. OBD- II port interconnects to various other ECUs making the vehicle vulnerable to attack if such a port is not secured. Self-driving vehicles are required to get 'smarter' in the near future in terms of security of the electronic parts (ECUs).

References

- [1] Car Hacking: study shows over 100 models at risk .[online], Available: <http://www.autoexpress.co.uk/car-news/consumer-news/92304/car-hacking-study-shows-over-100models-at-risk>
- [2] C. Miller and C. Valasek, "Adventures in Automotive Networks and Control Units," *illmatics.com/car_hacking.pdf*, (2013).
- [3] Can Your Car Really Be Hacked? Six Points to Know. [online]: <http://blog.caranddriver.com/can-your-car-really-be-hacked-six-points-to-know/>
- [4] S Bayer, R Jung and M Wolf, "OBD = Open Barn Door? Security Vulnerabilities and Protections for Vehicular On-Board Diagnosis (OBD)",
- [5] https://www.escrypt.com/fileadmin/escrypt/pdf/Whitepaper/OBD_Open_Barn_Door_Security_Vulnerabilities_and_Protections_for_Vehicular_On_Board_Diagnosis.pdf
- [6] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson and H.a.o. Shacham, "Experimental security analysis of a modern automobile", Security and Privacy (SP), 2010 IEEE Symposium on, (2010).
- [7] <http://resources.infosecinstitute.com/car-hacking-safety-without-security/>
- [8] How Modern Cars can be hacked. [online]: <http://www.tested.com/tech/concepts/461054how-modern-cars-can-be-hacked/>
- [9] CAN YOUR CAR REALLY BE HACKED? SIX POINTS TO KNOW. [ONLINE] : <http://blog.caranddriver.com/can-your-car-really-be-hacked-six-points-to-know/>
- [10] HACKERS CUT A CORVETTE'S BRAKES VIA A COMMON CAR GADGET. [ONLINE]: <http://www.wired.com/2015/08/hackers-cut-corvettes-brakes-via-common-car-gadget/>
- [11] 6 Wireless Ways to Hack into a car. [online]: <http://www.autoline.tv/journal/?p=34461>
- [12] [11] Securing the Connected Car – Ixia [online]: https://www.ixiacom.com/sites/default/files/resources/whitepaper/securing_the_connected_car.pdf
- [13] WHAT IS THE OBD PORT AND WHAT IS IT USED FOR? [ONLINE]: <http://www.makeuseof.com/tag/obd-port-used/>
- [14] "Comprehensive Experimental Analyses of Automotive Attack Surfaces", Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage, University of California, San Diego
- [15] S. Bayer, T. Enderle and P. Vyleta, "Demonstration of Secure On-Board Diagnostic," 2014

- [16] International Organization for Standardization (ISO), "ISO 11898: Road vehicles - Controller area network (CAN)," (2003).
- [17] a complete guide to hacking your vehicle bus on the cheap & easy – part 1 (hardware interface) [online] : <https://theksmith.com/software/hack-vehicle-bus-cheap-easy-part-1/>
- [18] "Secure Vehicular Communication Systems: Implementation, Performance, and Research Challenges", Kargl, Papadimitratos, Buttyan, Muter, Schoch, Wiedersheim, Thong, Calandriello, Held, Kung, Habaux, <http://icapeople.epfl.ch/panos/sevecomcomm-mag-2.pdf>
- [19] Y. Zhao. Telematics: safe and fun driving. Intelligent Systems, IEEE, 17(1):10–14, Jan/Feb 2002.
- [20] http://opengarages.org/handbook/2014_car_hackers_handbook_compressed.pdf
- [21] Cybersecurity for the Connected Vehicle [online]: https://www.capgemini.com/resource-file-access/resource/pdf/cybersecurity_for_the_connected_vehicle_pov.pdf
- [22] Building Comprehensive Security into Cars. [online]: http://www.symantec.com/content/en/us/enterprise/other_resources/building-security-into-cars-iot_enus.pdf
- [23] The Progressive Snapshot OBD-II dongle is vulnerable to security attacks. [online]: <http://telematicswire.net/the-progressive-snapshot-obd-ii-dongle-is-vulnerable-to-securityattacks/#icxSC6cbluUX5tzD.99>

Authors



Aastha Yadav, is a student, studying B.Tech Computer Science and Engg at VIT University, Vellore, Tamil Nadu, India. Her research interests include Knowledge based Artificial Intelligence and Network Security. She is a computer enthusiast and has keen interest in software development and management. She loves working on projects with real life application and scope using computer applications.



Gaurav Bose, is a student, pursuing a Bachelors of Technology in Computer Science and Engineering at VIT University, Vellore, Tamil Nadu, India. He has high ambition with zeal and aptitude for result oriented hard work. He is a computer enthusiast and his academic interests include Algorithmic analysis, Computer Networks, Theory of computation and IOT projects. He also has a passion for programming and has participated in multiple Hackathons and other competition. Gaurav is a team player and has a passion for projects that have a real life applications and use.



Radhika Bhange, is a student pursuing B. Tech Electronics and Communication engineering at Vellore Institute of Technology (VIT), Vellore, Tamil Nadu, India. Her academic interests include Digital Logic design, VLSI systems, Digital Signal Processing and Data structures and Algorithm. She is currently working with the IEEE Signal Processing Systems, VIT Student chapter, Vellore. She is an avid reader and computer software enthusiast. She has a passion for programming and loves working on IOT projects.



Karan Kapoor, is a student pursuing Bachelors of Technology in Mechanical specialization in Automotive Engineering at Vellore Institute of Technology (VIT) University Vellore, Tamil Nadu, India. He is a mechanical maniac where his autonomous interest regarding “power-train” Department, Automotive Electricals and Electronics. He likes to discover things explore them in various fields of Automotive Transmission. Karan has proved his potent in various competitions as a team player and a team leader in sweepstakes like “Shell Eco-marathon” (Asia).



Ronnie D. Caytiles, He had his Bachelor of Science in Computer Engineering- Western Institute of Technology, Iloilo City, Philippines. He finished his Masters and Ph.D. in Multimedia Engineering, Hannam University, Daejeon, Korea. Currently, he serves as an Assistant Professor at Multimedia Engineering department, Hannam University, Daejeon, Korea. His research interests include Mobile Computing, Multimedia Communication, Information Technology Security, Ubiquitous Computing, Control and Automation.



N. Ch. S. N. Iyengar, he is a Professor, SCS Engineering at VIT University, Vellore, TN, India. His research interests include Distributed Computing, Information Security, Intelligent Computing, and Fluid Dynamics (Porous Media). He had much teaching and research experience with a good number of publications in reputed International Journals & Conferences. He chaired many Intl. Conf. delivered Key note lectures, served as PC Member/Reviewer. He is Editorial Board member for many Int'l Journals like *Int. J. of Advances in Science and Technology*, of SERSC, *Cybernetics and Information Technologies* (CIT)-Bulgaria, Egyptian Computer Science Journal -Egypt, IJCA & IJConvC of Inderscience -China, *etc.*, Also Editor in Chief for International Journal of Software Engineering and Applications(IJSEA) of AIRCC, Advances in Computer Science (ASC) of PPH, Guest editor for "Cloud Computing and Services" IJCNS.

