

A Balanced Approach for Securing the OBD-II Port

Tom R. Markham and Alex Chernoguzov
 Honeywell

ABSTRACT

The On-Board Diagnostics II (OBD-II) port began as a means of extracting diagnostic information and supporting the right to repair. Self-driving vehicles and cellular dongles plugged into the OBD-II port were not anticipated. Researchers have shown that the cellular modem on an OBD-II dongle may be hacked, allowing the attacker to tamper with the vehicle brakes. ADAS, self-driving features and other vehicle functions may be vulnerable as well. The industry must balance the interests of multiple stakeholders including Original Equipment Manufacturers (OEMs) who are required to provide OBD function, repair shops which have a legitimate need to access the OBD functions, dongle providers and drivers. OEMs need the ability to protect drivers and manage liability by limiting how a device or software application may modify the operation of a vehicle. This paper outlines a technical approach based upon cryptographic authentication and granular access control policy which addresses the needs of stakeholders. This allows the OEM to protect the security of the vehicle by carefully controlling the functions a particular device plugged into the OBD-II port is able to perform. This allows device makers (diagnostic tools, insurance dongles, etc.) to have their products certified to work with the OEM's vehicles. The result is the OEMs can protect driver safety and maintain the right to repair.

CITATION: Markham, T. and Chernoguzov, A., "A Balanced Approach for Securing the OBD-II Port," *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.* 10(2):2017, doi:10.4271/2017-01-1662.

1. INTRODUCTION

Motivation

There is a real and growing threat to automotive cyber security and, in turn, safety. In particular, the OBD-II port has become a greater risk [19]. The number of vehicles with brakes, steering and acceleration influenced by messages on the CAN bus is increasing due to self-parking, ADAS and the move toward autonomous driving. The OBD-II port has transitioned from a physical port only accessible from inside the cabin of the vehicle to one that is wirelessly accessible via cellular modems, Bluetooth and Wi-Fi dongles [9, 1]. There is evidence that interest in the cyber security of vehicles is migrating from white hat researchers to hackers with hostile intent. Taken together, these facts call for taking a fresh look at the OBD-II port and identifying an up-to-date approach for protecting vehicle security and safety.

Related Work

There has been a growing list of researchers who have demonstrated that gaining access to the CAN bus via the OBD-II port can allow a hacker to influence control of a vehicle [8]. Those researchers who have used wireless access via an OBD-II device to compromise a vehicle [12, 3] are of particular interest.

Prior Society of Automotive Engineers (SAE) standards work in the form of SAE J2186 - E/E Data Link Security [14] addressed access to ECUs via the Data Link Connector (DLC). (Throughout this document "OBD" is used as a shorthand for OBD-II and the DLC.)

In some sense, J2186 may have been ahead of its time because when it was revised in 2005 the threat environment and motivation to secure vehicles was very different from today. The OBD-related threat has recently gained the attention of Congress which in a Sep 12, 2016 letter to NHTSA [13] wrote "... stakeholders cited the ports and the direct connection they provide to the vehicle internal networks as one of the fundamental sources of cyber security risk in the modern vehicle ecosystem."

Within the broader cyber security community there has been a great deal of work done on cryptographic protocols [11], public key cryptography [16], public key infrastructure and role based access control [17, 10, and 2]. These techniques and lessons learned can accelerate efforts to protect vehicles from attacks through the OBD port.

The remainder of this paper is organized as follows:

2. Brief history of vehicle network technologies, providing background on how we got here.
3. Requirements, informally summarizes the cyber security challenges this paper is addressing. There are many other cyber security items to address beyond the limited scope of this paper.
4. Technical approach, provides one technical approach for meeting the needs of various stakeholders in a manner which is scalable, cost effective and secure.
5. Operation, outlines how the technical approach would be used for device certification and use.

6. Design considerations, discusses certification, implementation cost and technology options.
7. Summary/Conclusions, touches upon standardization of the OBD security concept and application to the telematics control unit (TCU).

2. BRIEF HISTORY OF VEHICLE NETWORK TECHNOLOGIES

CAN Bus

What was the original environment for the CAN bus? Historically the CAN bus was an island. An air-gapped network contained within the sheet metal envelope of the vehicle. Everything on the network was integrated by the OEM and trusted to preserve the cyber security of the vehicle. Anyone attempting to send CAN bus messages on the bus was assumed to be inside the vehicle and therefore trusted.

This level of trust and openness on the CAN bus was reasonable back in the day when brakes, steering and transmission were controlled by physical linkages, not a network. Today the lack of source authentication on the CAN bus is a security and safety issue.

OBD

What was the original purpose of the data link connector and OBD? It was designed to allow repair shops to connect with engine controls to perform diagnostics and gather emissions related information. Few envisioned the OBD port as a means of remotely injecting commands which could affect brakes, steering or acceleration. When J1979 was first issued in 1991, who imagined an insecure cellular modem connected to the vehicle OBD port controlling vehicle operation?

This paper and SAE J2186 both address aspects of OBD security. However, SAE J2186 was focused on protecting the integrity of a single Electronic Control Unit (ECU). i.e. J2186 could be used to prevent a device from changing firmware or calibration parameters within a specific ECU unless the device at the OBD port could authenticate to the ECU as a device authorized to make changes to that ECU. While this proposal can also prevent an OBD connected device from making changes to an ECU it can also prevent the OBD connected device from injecting CAN messages which trigger ECU actions. e.g., a message injected via the OBD triggers an ECU to unlock the doors.

The CAN bus does not perform source authentication of messages placed on the CAN bus. Thus, an attacker who is able to inject messages into the OBD port can send messages which appear to come from one of the ECUs on the bus. Referring to Figure 1, if the OBD security module was not present, the OBD connected device would be able to place messages on the CAN bus which would normally be sent by ECU_1.x or ECU_2.x. Thus, anything these

ECUs would normally do could be done by an attacker via the OBD. A few of the many things the attacker could do via the OBD port include:

- Flash the headlights
- Unlock the doors
- Turn the air conditioning on/off
- Activate anti-lock brake functions
- Turn the front wheels (on vehicle with automatic parking capability)
- Change the speed shown on the speedometer

Some of these actions may be categorized as nuisance aimed at distracting the driver. Other actions directly affect vehicle operation taking control away from the driver or interfering with driver actions. Either can result in a safety incident.

OBD Devices

The OBD-II port is a source of data that can be used and manipulated for insurance tracking, monitoring vehicle performance and other “diagnostic” functions. Low cost (less than \$20) wireless OBD devices are commercially available [1]. These can provide access via cellular, Wi-Fi or Bluetooth. The security of these devices is questionable. The web description for one model makes the claim

Easy to Use: Plug the device in your car's OBD2 port, turn on your car, enable Bluetooth on your Android phone or tablet, search for "OBDII" and pair with it (pin 1234), run the download App with simple settings and wait until it connects your car's ECU successfully

The hackers now know the pairing code for all devices of this model. What other security flaws could be lurking in these devices?

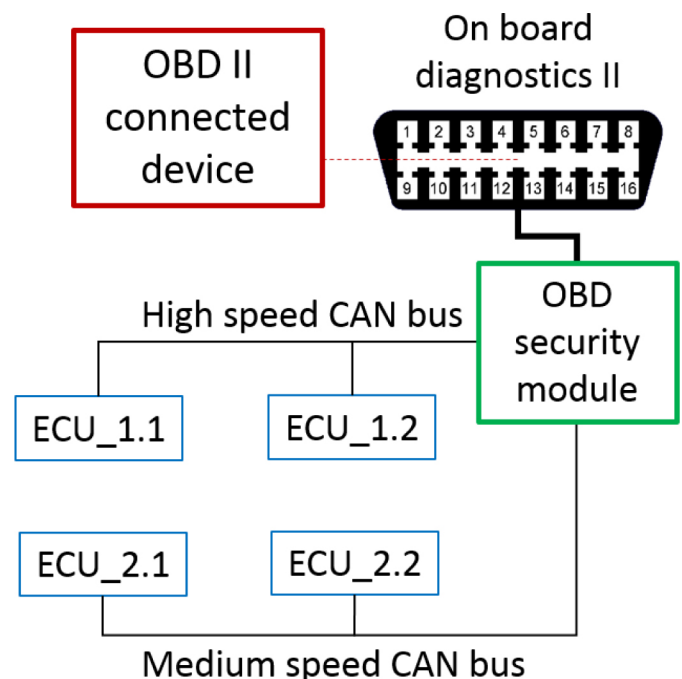


Figure 1. Placement of the OBD security module within the vehicle

Hacking Culture

The world of hacking has also changed over the last 20 years. We have witnessed a progression from hacking a PC via a virus on floppy disc to hacking a website to get bragging rights and more recently hacking of industrial control systems. The Stuxnet attack on an industrial control system [7] was a wake-up call: hacking could remotely damage or destroy physical systems. This, of course, has been followed by multiple demonstration hacks of vehicles as well as research addressing vehicle security [18]. What is more concerning is the change in who is doing the hacking and their motivation. Years ago, bored college students hacked to show their skills. Today organized crime hacks for money. Ransomware attacks on home PCs and even businesses are common. When will we see ransomware against vehicles?

3. REQUIREMENTS

The OBD environment has been examined to identify security requirements necessary to protect vehicle safety. It is obvious that a one-size-fits-all access control policy is not appropriate. The requirements to secure the OBD port and associated components are outlined below:

- Support diagnostics. The OEM is required by law to provide an OBD-II port to support diagnostics. Independent repair shops and vehicle owners have a right to repair which includes access to many vehicle functions via the OBD-II. Some “secured Functions” are unlocked per SAE J2186 but many functions are not locked.
- Provide third parties (e.g. insurance companies) read access to vehicle data such as Vehicle Identification Number (VIN), speed and braking via OBD.
- Enable OBD device providers the ability to offer vehicle-related services via the OBD port. The industry must balance access with vehicle security and safety concerns.
- Provide the vehicle owner the ability to use mobile apps (Torque, Dash, OBD Auto Doctor, etc.) to retrieve information from their vehicle.
- Deliver vehicles which are safe. The safety of modern automobiles requires that they have cyber security to protect them from malicious commands/data injected via the OBD port. This implies these derived requirements.
 - Provide a means to test and certify the CAN messages injected by an OBD connected device (e.g., Insurance dongle, diagnostic tool).
 - Provide a means to authenticate a device plugged into an OBD port. i.e., determine the manufacturer of the device and the model number.
 - Determine and enforce the OEM approved access rights for the authenticated device relative to the OBD port. e.g., the device can read all data but only write to the body control modules.

4. TECHNICAL APPROACH

This section provides an introduction to the technologies which can be used to solve the OBD security issue. It introduces key technologies and outlines the system.

Defensive Technologies

Fortunately, there are defensive technologies which can be employed to enhance security. Public key cryptography and access control mechanisms could be combined to provide an authenticated access control function for the OBD port.

Public Key

Public key (aka asymmetric key) cryptography [16] technology in the form of RSA and elliptic curve makes it possible to provide digital signatures and create public key infrastructure tailored to the needs of protecting vehicles. This is a mature technology already in use in other industries.

Public key cryptography uses two keys which are mathematically-related. The private key must be protected from unauthorized access. It is typically stored in some form of protected hardware and only accessed by the cryptographic engine within the hardware. Thus, the private key is used by the cryptographic engine to sign and encrypt/decrypt data but the key cannot be extracted from the cryptographic engine. The corresponding public key may be freely distributed. It does not require any confidentiality protection. The approach described in this paper makes use of two cryptographic functions.

A *digital signature* is an operation in which the creator digitally signs a piece of data by using the private key. The signature is a string of bits which practically speaking, could only be created by the entity holding the private key. Anyone holding the corresponding public key is then able to use that public key to perform a cryptographic operation and confirm that the piece of data in question was indeed signed by the entity holding the private key. As an example, if an OEM wanted to sign a firmware file such that any vehicle could verify that the firmware came from the OEM, the OEM would:

1. Create the file containing the firmware.
2. Compute a hash function over the firmware
3. Use the OEM private key to encrypt the hash result, thus creating a digital signature.
4. Send the original firmware file and the encrypted hash result together to any device which is to use the firmware.

The receiving entity, which has obtained the OEM public key through a trusted channel (e.g. loaded into the vehicle at the time of manufacture) then performs the following:

1. Compute the hash function over the firmware
2. Use the OEM public key to decrypt the encrypted hash result distributed with the firmware file.
3. Compare the locally computed hash result to the result of the decrypted hash from step 2.