Securing Connected Vehicle through Secure Product Development

A connected vehicle brings in an interesting viewpoint. It is conceptually viewed as another node in a network. For consumer, it is just another extension to their already connected devices at home or work. However, this connectivity introduces disturbing security risks.

Connected vehicles introduce a different security paradigm. Until recently, a vehicle had been considered to be a self-contained system with wired, peer-to-peer network connecting its equipments. The design use cases and assumptions were based on the fact that the connectivity of equipments would be bound to a local system. The constrained environment allowed the designers to provide only basic security measures. Getting physical access to the vehicle was a pre-requisite to launch an attack. Due to this, the attack surface was small. With the vehicle connected to internet or a larger communication network, it becomes another node of the network and its attack surface increases. So, the assumptions about possible misuse changes and as a result, the security controls are now deemed inadequate in dealing with newer threats making a vehicle vulnerable.

For example, an attacker who could connect remotely to a vehicle's infotainment system could find a way to inject command targeted to a steering control ECU which is a safety critical system. If successful, she would be able to control the steering wheel. Another example could be the tyre pressure measuring sensor connecting with the dashboard to show the reading. If that communication could be intercepted and a wrong data injected, the driver would see the tyre pressure to be okay whereas in reality it is very low which at certain speed and on a certain surface could cause fatal accident.

The security of a connected vehicle closely resembles mobile security, IoT security and application security combined. Out of the three, IoT security is relatively new and has similar challenges as connected vehicles. So, the approach towards securing connected vehicle will require a framework which will adapt the processes, techniques, technologies and best practices from the three fields mentioned above. The cyber security processes for all three recommend the use of secure product development lifecycle. Secure product development mandates implementing security practices from ideation to deployment. For connected vehicles, the same should be applied. Security must be considered when an OEM is designing a vehicle. All the subsystem and component design must have their respective security requirements identified and confirmed at the design time. The subsequent lifecycle stages must have necessary checks and balances to ensure that the security requirements have indeed been developed. At various milestones, there must be an assurance process which will validate the security of the implemented functions and if the standards are not met, will feed the gaps back to the design and development teams. At the end, a final assurance exercise must happen for go/no-go decision.

Let us perform a simplified, scaled down but representative threat modelling to establish the need of a process like this.

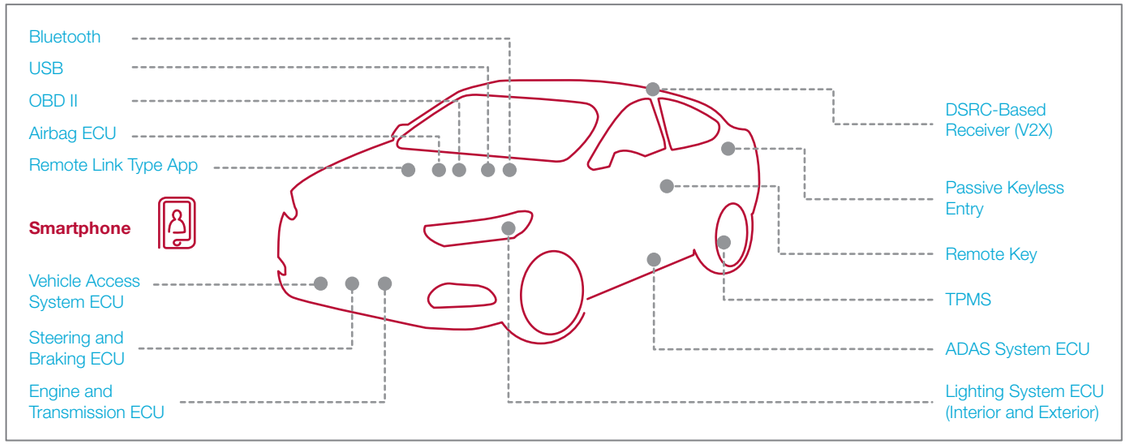

Figure 1 : Remote Attack Surfaces of a Connected Vehicle

The success of a hack depends on three main parameters:

- Remote attack surfaces
- Cyber-physical features
- In-vehicle network architecture

As per Charlie Miller and Chris Valasek, 20% of the models (year 2014-15) from different manufacturers are vulnerable to more than seven categories of remote attack. (Miller and Valasek are security researchers who are famous for their successful remote hacking of FCA Cherokee Jeep)[1].

Threat Agents

- **Researchers and Hobbyists:** Universities, government labs, defense labs. Motivations are usually positive, to study and conduct research.
- **Pranksters and Hacktivists:** Take opportunity to demonstrate their skills or promote their cause but with negative outcomes for the product owners and manufacturers.
- **Owners and Operators:** Many car hacking tools exist with owners and often they want to hack their own vehicles to improve performance, to bypass restriction set by manufacturers or regulators or disable components to obfuscate their fraudulent actions.
- **Organized Crime:** Has always been a threat to vehicles. Main motivation is financial gain. They can launch DoS attack, drop malware, ransomware. In recent time, it has been found that there is a commercial service provided by organized crime groups known as “Cyber-crime-as-a-Service”.
- **Nation States:** Motivations can vary widely. May include (but not limited to) industrial espionage, surveillance, economic and physical warfare, intervention to assist national manufacturers against foreign competition, tracking and audio monitoring of high-value objects.
- **Transportation Infrastructure:** Next-generation car V2V communication. Security and safety issues can occur through attacks and misbehavior of the surrounding infrastructure.
  Example: manipulation of traffic lights to confuse smart cars causing accidents.

[1] *http://www.darkreading.com/vulnerabilities---threats/this-time-miller-and-valasek-hack-the-jeep-at-speed/d/d-id/1326468*

Threat Model

Given the internal network depicted in Figure 1, and the external connected devices, the threats can be classified using Microsoft STRIDE and SAE Impact methodologies. An example is shown in Figure-2.

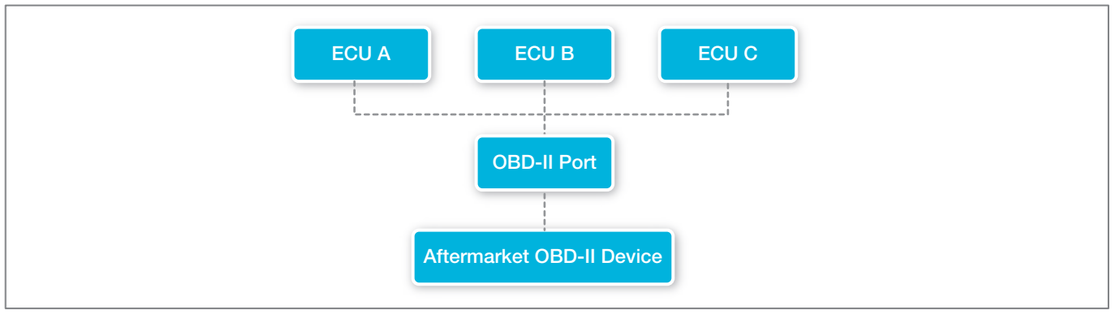

Figure 2 : ECUs connected to a single CAN Bus

Vulnerability	ECU Affected	Comments	Microsoft STRIDE						SAE Impact			
			S	T	R	I	D	E	S	P	F	O
Hardcoded Credentials	None							✓	S0	S3	S0	S0
Arbitrary Command Injection	OBD Connected Bus					✓			S4	S4	S0	S0
Arbitrary CAN Injection	OBD Connected Bus	Full Device Compromise	✓	✓	✓	✓	✓	✓	S3	S3	S3	S3

Table 1 : Threat classification using STRIDE and SAE Impact

*** SAE Impact divides threats into four areas: S = Safety, P = Privacy, F = Financial, O = Operational*
*** For each area, impact classes are rated from S0 to S4*

Potential Risks

- **Safety-Critical Risks:** Driver distractions (e.g. sudden unexpected volume, wipers activation), engine shutoff or degradation, steering changes (autonomous vehicles).
- **Less Safety-Critical Vehicle Specific Risks:** Theft of the car or contents, enabling physical crime against occupants, insurance or lease fraud, eavesdropping on occupants, theft of information (e.g. personal profile, phone list), vector for attacking mobile devices in the car, theft of PII, tracking the vehicles, location.

Key vulnerabilities found in a connected vehicle

- Insecure firmware updates and downloads
- Hardcoded or non-existent Bluetooth PIN
- Weak WPA2 password
- Hardcoded credentials
- Internet-enabled administration interface

Important attack vectors

- Arbitrarily modify firmware
- Maliciously update remote firmware
- Lock/unlock doors
- Turn on/off vehicle
- Affect vehicle GPS tracking, speed, heading and altitude
- Read the car’s internal data – temperature, fuel levels, diagnostic trouble codes etc.
- Inject arbitrary CAN packet

Common Architecture Issues

- The primary processor is a simple processor. It can convert External Network Protocol to CAN and vice versa. The logic is implemented in upstream systems and it does not include any security functionality e.g. authentication, command validation etc.
- Due to absence of traffic filtering at device and OBD-II port security is completely dependent on perimeter i.e. external network interface whose strength varies. WPA2 with password of lower strength is easy to break. Equally easy is to brute force Bluetooth PIN. Many Bluetooth PIN are widely shared and many a times the default (0000) is used for long time.
- Component manufacturers put in undocumented features at the time of development and either intentionally or by mistake do not take it out. The thought process behind it is security through obscurity which historically has been found to be unsuccessful in protecting an asset.
- Insecure firmware upgrade is a key issue which can introduce a significantly large attack surface.

Recommendations

Hardware Security: Use of secure boot and software attestation function, trusted platform module, tamper protection, cryptographic accelerator, active memory protection, device identity directly on device (e.g. Intel EPID, physically uncloneable function)

Software Security: Secure boot, partitioned OS, module level authentication, enforcement of approved and appropriate behavior, secure product development lifecycle

Network Security: Message and device authentication, Identify and enforce predictably holistic behavior, access control

Cloud Security: Secure authenticated channel to cloud, Remote monitoring of vehicle, threat intelligence exchange, OTA updates, credential management

Supply-chain Security: Authorized distribution channel, track and trace components, continuity of supply, ability to identify uncertified component.

About the Author:

Arnab Chattopadhyay is the cyber security leader within the Product & Engineering Services business unit at Capgemini. He has over 23 years of experience with expertise to provide solutions to complex problems in the area of IT Security. Arnab advises our clients to secure their cyber physical systems, manage cyber security risks right up to the product hardware level and adhere to compliance requirements.
You can contact Arnab at arnab.chattopadhyay@capgemini.com.

For more information,write to us at
marketing.pes.in@capgemini.com

About Capgemini

With more than 190,000 people, Capgemini is present in over 40 countries and celebrates its 50th Anniversary year in 2017. A global leader in consulting, technology and outsourcing services, the Group reported 2016 global revenues of EUR 12.5 billion. Together with its clients, Capgemini creates and delivers business, technology and digital solutions that fit their needs, enabling them to achieve innovation and competitiveness. A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

Learn more about us at
www.capgemini.com