University of Western Australia

Final year project

Evaluating Attack Risk on Connected Vehicles

Terence Leong

School of Electrical, Electronic & Computer Engineering

Supervised by

Dr. Jin Hong

Faculty of Engineering, Computing & Mathematics

2 May 2020

Contents

1 Introduction

In recent decades, modern vehicles have been rapidly increasing their number of in-vehicle electronic devices and ever increasing complexity of in-vehicle systems. Therefore it has been imperative to rely upon external network systems for efficient communication services. However, this development in technology has exposed them to new types of internal and external threats. As a result, cyber security has become one of the most significant challenges in connected vehicular systems [1], [2]. The biggest security concern relates to the in-vehicle network composed of Electronic Control Units (ECU) and various buses such as the Control Area Network (CAN) bus, which are primarily used to monitor and control the state of vehicular systems [3]. A variety of attack types against the ECU and buses have been discovered over the years and a number of defence techniques have been developed to combat them.

In 2010, Koscher et *al.* were the first group of researchers to perform an attack on a real car [4]. They were able to successfully control a range of modules in two 2009 automobiles of the same make and model by taking advantage of the vulnerabilities presented by the CAN bus through frame injection. In 2015, Chrysler was forced to recall 1.4 million vehicles due to a pair of hacker demonstrating that it was possible to remotely hijack a Jeep's digital system over the internet [5]. As those examples demonstrate, a more proactive approach has to be taken to efficiently react to these attacks. In a recent survey study, eight common attack patterns were identified and classified including Denial of Service Attacks (DoS), black hole attacks and replay attacks [3].

Extensive research attention has been paid to develop solid defences against these attacks. Defences can be categorised based on their approach: crytographic defence, network security defence, software vulnerability detection defence and malware detection defence [2]. The most robust and highly researched area involves cryptography with defence techniques covering all the eight common attack patterns. However, limitations still exist for these techniques. Some of these limitations include high overhead, difficult implementation and low processing speed.

Even though there exist a long-established computer security policy, the industry standard does not follow it due to hardware constraints and differences in network configuration for these in-vehicle systems [6], [7], which further increases the difficulty to develop an all-encompassing defence.

The goal of this research is to evaluate the risk and impact of attacks in vehicular systems. I first investigate a variety of common attack patterns on in-vehicle systems and a number of different defence implementations suggested over the years. Each suggested defence is then mapped to the attack pattern that it would prevent. Next, I explore intrusion detection system (IDS) in vehicular systems and identify techniques that can actively intervene when an attack occurs. Finally, I discuss the chosen approach to implement a robust system that can evaluate the impact of attacks on vehicular systems by developing models and metrics to quantitatively compare impact of different attacks to optimise security.

2 Literature Review

2.1 Vehicular Security Technology

2.1.1 Vehicular System Architecture

The central communication point for an in-vehicle system is the Electrical Control Unit (ECU). The ECU is an embedded device that was designed to monitor vehicle state and control multiple systems within the overall architecture [8], [9]. Due to the number of increasing features in a vehicular system, the ECU was developed as multiple components each controlling a different set of functions within the system. This meant that a network had to be developed for seamless communication between ECU components.

Nowadays, various networks exist which provide efficient communication amongst control modules within a vehicle. Each of these networks help provide efficient communication and strengthens the overall performance of the vehicle. The most commonly found networks are the Controller Area Network (CAN), FlexRay, Media Oriented System Transport (MOST) and Local Interconnect Network (LIN) [8], [10], [11]. The networking type that I'll be focusing on is the CAN bus. This is because the CAN bus is the most widely used protocol for in-vehicle networks, allowing different ECU components to communicate with each other through a wired CAN bus. In addition, the CAN bus is heart of a connected vehicular system, therefore securing it would greatly improve a vehicular system's defence capabilities.

2.1.2 Controller Area Network

The CAN bus is a robust linear-type network designed to allow devices to communicate with each other without a central master that controls all the devices.Cabling for a CAN bus is low in cost and due to the linear nature of this network, it makes it easy for new nodes to connect to it [8]. However, the CAN bus was designed as a closed network and has several vulnerabilities such as:

- Multicast Messaging: Messages are broadcasted to everyone. Passive attack-

ers are capable of listening in on this broadcast with ease.

- No authentication: There is no way to ensure that messages received are from a credible source. Attackers are able to perform attacks by pretending to be a CAN bus.
- No encryption: Data on a CAN bus is not encrypted. Attackers can easily analyse a CAN frame.
- Common point of entry: Once a CAN bus has been compromised, it is possible to access all parameters on it without limit.

Despite that, the CAN bus only defines the protocols for the physical and data link layers within an OSI model. Hence, error checking and addressing are not implemented for it. The network layer is not included within a CAN protocol because it would require more infrastructure for the routing process in addition to the higher cost investment compared to a data link layer approach. Lower latency is also highly preferred for a vehicular system, therefore a data link layer solution is more pragmatic [12].

2.2 Security Attacks

As the prominence of connected cars rise throughout the years, so has the variety of attacks that can be performed on them. These attacks can be analysed from two different perspectives: the attack surface and attack vector [1]. The attack surface is described as the total sum of vulnerabilities present in the network that is accessible to attackers. In this scenario, there exist an attack surface through the On-Board Diagnostic II (OBD-II) and firmware of the media player manufacturer. Whereas the attack vector is seen as the pathway taken by attackers to achieve a malicious outcome. This could include the potential exploitation of web applications designed for vehicles or spoofing certain commands to gain access to on-board devices. The following is the different attack vectors that are typically taken to exploit a vehicular system:

- Denial of Service Attacks (DoS): This form of attack involves flooding the network with large amount of packets in an attempt to overload it. This

would prevent the transmission and processing of legitimate incoming packets.

- Distributed Denial-of-service Attacks (DDoS): The basic concept of flooding the network is exactly the same as a DoS attack. However, due to the high strain placed on the attacker's resources in a DoS, DDoS was developed. As opposed to using a single attacking node, DDoS utilises multiple IP addresses to perform an attack. Thereby reducing resource strain dramatically.
- Black-Hole Attacks: This form of attack creates a metaphorical 'black hole' where no packets are able to move through the network. This type of attack causes serious damage to network performance and routing. A well placed black-hole node on a critical path could cripple the entire network.
- Replay Attacks: This form of attack attempts to capture information before relaying the packet to its destination. Replay attacks allow the attacker to use the intercepted information later to authenticate themselves or attack the network.
- Impersonation Attacks: This form of attack allows the attacker to gain access to information and resource that is otherwise restricted. Impersonation attacks are typically seen following a replay attack, as a replay attack would provide login details and passwords.
- Fuzzy Attacks: This form of attack is achieved by iterative injection of random packets. This allows the attacker to find vulnerabilities through unexpected values or random data entered into the network.
- Malfunction Attacks: This form of attack target specific functions within the network and cause it to stop working.

These large number of security threats can be performed through internal and external attacks. In terms of internal attacks, the attacker would typically exploit the vulnerabilities found in the OBD-II port. This would allow for eavesdropping on bus traffic or even frame sending. In addition, if an attacker is capable of controlling a single node in the network, current protocols make it possible for him to gain control over every ECU in the network. External attacks mostly refers

to the exploitation of a vehicle's communication interface, therefore it would not require physical access to the embedded network. These attacks are performed through indirect physical access, short range wireless access or even long range wireless access [13].

2.3 Defence Against the Attacks

An increase in connectivity for vehicular systems has proven to expose vehicles to more security and safety dangers, making safety critical features a huge concern in modern day research. Despite all research efforts, there is currently no singular conclusive method to defend against attacks.

Based on a recent survey study, security defences can be categorised into: cryptography, network security, software vulnerability detection and malware detection [2]. The most immensely researched of which is the cryptographic approach to authentication and integrity. Message authentication code (MACs) and digital signature are some prime examples of this. As a CAN frame can only accommodate up to eight bytes, it is difficult to implement a crytographic approach that is robust enough to prevent all attacks [14]. CAN+ attempts to solve this issue by using out of band channels to transmit bits instead. However a large drawback of this method and all similar ones remain in the form of backward compatibility [15]. Some have attempted to address this problem, but none of which provide the security necessary for this method to be reliable enough for practical use. Therefore it can be concluded that methods that rely solely on cryptography will be vulnerable to brute force attacks. Table 1 shows the mapping of the application for existing effective defence against common security attacks:

	Defence Categories			
Cyber-Attacks	Cryptography	Network Security	Software Vulnerability Detection	Malware Detection
DoS	x	x		
DDoS	x	x		
Black-hole	x	x		
Replay	x			
Sybil	x	x		
Impersonation	x	x		
Malware	x		x	x
Falsified Information	x	x		
Timing	x			

Table 1: Application of Defence Type to Cyber-Attack Type [2]

2.3.1 Intrusion Detection System (IDS)

An IDS is a system capable of monitoring traffic moving on networks and through systems for suspicious activity and issues alerts when such activity is discovered [16]. An example of this is an anomaly intrusion detection algorithm based on survival analysis proposed by Kim et *al.* [1] Anomaly detection is used to detect suspicious patterns that do not conform to the expected behaviour, whereas survival analysis is a statistical model used to discover survival rate and survival duration of measurement objects [1]. The proposed method can be split into two major components: chuck-based threshold measurement and the detection algorithm.

By applying survival analysis in this manner, the IDS is not restricted to searching for specific patterns, it is also capable of detecting unknown attacks and doesn't require regular updates of signature concerning attack patterns. However, the major disadvantage posed by this method is the fact that it is incapable of checking the condition of the vehicle or perform safety diagnostics either.

3 Problem Identification

Currently, IDS approaches have a variety of ways to tackle the detection of attacks on vehicular system. However, they incapable of determining the impact of these attacks making it impossible to check the condition of the vehicle or provide the appropriate measures to react to the situation. An example of this would be if a vehicle were to be exposed to two different attacks, one against the engine and one against the windscreen wipers, the system is incapable of informing the user which attack is more severe or require immediate attention. This would pose a major issue as the user would be incapable of managing their resources optimally to deal with the situation. The main objectives of this project are to:

- Evaluate impact of different attacks against vehicular system through attack simulations in the CAN protocol
- Develop a risk impact assessment and prioritization algorithm
- Evaluate the effectiveness of the risk impact assessment and prioritization algorithm for determining the effectiveness of countermeasures against attacks

4 Methodology

The purpose of this project is to implement of an algorithm capable of quantifying the risk presented by an attack, evaluating its final impact on the vehicle and selecting the most optimal countermeasure against the attack. This will be a key component of advancing security for vehicular systems. The project can be split into three main sections: Classification of attacks,

4.1 Classification of Attacks

Utilising the extensive list of attack patterns, we can now utilise threat modelling methods to inform defensive measures. This will help us determine which attacks to simulate against our vehicular system. The threat modelling method selected for this project has to fulfill a few simple conditions:

- Easy to use
- Capable of capturing all or most threats
- Reliable
- Proven to be effective

In the paper by Shevchenko et *al.* [17], 12 different threat modelling techniques were surveyed, including STRIDE, security card and Persona non Grata [17]. STRIDE is an acronym that stands for six categories of security risk: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of service and Elevation of Privileges [15], [18]. STRIDE is capable of systematically examining and exposing gaps in the security posture of any application. Each of these categories aim to address one aspect of security. Security card moves away from the checklist approach of STRIDE and inspires more brainstorming techniques. The motivation behind this model is to help users identify more complicated attacks. Security card was seen to be producing more effective results, but also more false positives compared to STRIDE [17].

Persona non Grata approach makes threat modelling more tractable by asking

users to focus on the motivation and abilities of attackers. Users are to think and act like an attacker to better understand the direction an attack can come from. This method produces less false positives compared to STRIDE, but its also incapable of capturing all the threats due to a much narrower perspective on the system [19]. Other threat modelling techniques were considered as well, but STRIDE was ultimately the choice due to its maturity and simplicity as a model. The STRIDE model will be used to determine the relevant elements as seen in table 2:

ELEMENT	S	T	R	I	D	E
ECU	x	x	x	x	x	x
Data Flow		x		x	x	
External Entity	x					

Table 2: STRIDE categories mapped on Relevant Interconnections [20]

The relevant STRIDE classes here was used by Winsen to identify and determine the severity of various attacks against a vehicular system [20]. The exact same table can be applied to our research for risk evaluation and metric selection.

4.2 Generating Attack Data

In order to generate the necessary data, a synthetic attack generator has to be implemented. A synthetic attack generator will be implemented as both an emulated attack and simulated attack type for this. An emulated attack will replicate the function of attacking a vehicle. A simple implementation of an emulated attack involves looping and sending a function through an Arduino acting as a CAN sniffer to a vehicular system. The limitations of this approach is its flexibility, as it won't be able to emulate more complex attacks. On the other hand, the simulated attack will attempt to reproduce the internal state of an attack allowing for more intricate designs. An Arduino will be used as a CAN sniffer. The purpose of this synthetic attack generator is to generate data of what the CAN bus sequence would look like during an attack. CAN message ID will encompass most of the raw data that we collect. This will allow the IDS to differentiate between a normal driving pattern and an attack pattern. This is of the utmost importance as it forms the basis of how well our IDS will be at detecting attacks. The better the IDS is at

detecting and differentiating between different attack patterns, the more complex and robust the risk evaluation process can be.

In the context of this project, the OBD-II will be required as a physical connector for the Arduino to gain access to the internal CAN network of the system. A Python CAN monitor tool will be used for analysing CAN bus data collected through the Arduino. The major limitation here is that not all vehicular systems allow unrestricted access to all the internal CAN networks through the OBD-II. Another limitation that arises comes from raw data processing. Reading and decoding raw data such as CAN message ID from the Arduino will be difficult. This is because the CAN ID collected does not inform us which one performs which function, hence manual cross referencing of CAN ID and function will be required for this process.

4.3 Evaluating Attack Risk

The evaluation of attack risk is heavily reliant upon the metrics that are chosen to determine them. The selected metrics for our project has to test for vulnerabilities whilst also taking into account limitations such as cost efficiency and CAN bus capabilities. These metrics will act as the framework for the risk assessment and prioritization algorithm. The general idea behind this is to create an algorithm capable of detecting unnatural vehicle behaviour, determining the type of attack it is and returning a risk analysis score to the user. It is necessary to evaluate the risk behind an attack to provide the proper assistance and action to be taken.

In its most simple form, risk can be expressed by the product of two parameters: the likelihood of an attack and the impact of the threat when it materialises [21]. Therefore we need to select a set of metrics that will help gauge the likelihood of an attack occurring and another set to determine the severity of it. Metrics have to be applicable in the context of a CAN bus. This means that metrics that test authentication, integrity, availability and access control will be selected due to them being the most vulnerable and exploitable features within a CAN bus.

Hughes and Cybenko [22] states that cyber security vulnerabilities can be modelled on three tenets: The existence of inherent system susceptibilities, the attacker's access to the susceptibility and the attacker's capability to exploit the susceptibility [22]. This can applied to our project as it'll test all four features of the CAN bus. However, the limitation presented by these quantification metrics is that it'll only evaluate the likelihood of a potential attack, but not the impact of the attack. If this approaches were to be paired with another set of metrics for determining severity, it might prove to be fairly effective. In a seperate paper, Abraham and Nair [23] list a set of classes that network security metrics can fall under using attack graphs. The most relevant of which is the time-based metrics. These metrics quantify how quickly a network can take preemptive measures to respond to attacks [23]. This could potentially be used as a metric for determining severity.

A different set of metrics proposed by Moukahal and Zulkernine [24] also aims to identify weak components in a connected vehicle through measuring the security vulnerabilities in every functionality. The five metrics they chose were ECU coupling risk, communication risk, complexity risk, input and output data risk, and history of security issues. The problem with this is that the cost of testing vulnerabilities could potentially be inefficient, hence it would not be approved by car manufacturers [24].

5 Timeline

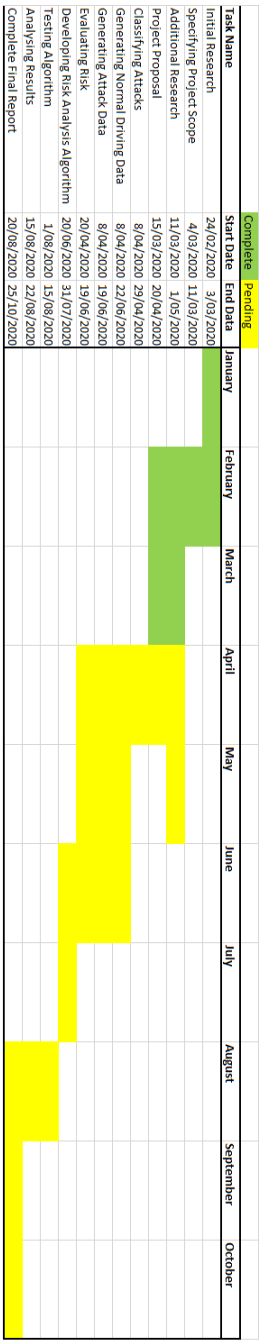

Figure 1: Thesis Gantt Chart

References

[1] M. Han, B.-I. Kwak and H. K. Kim, 'Anomaly intrusion detection method for vehicular networks based on survival analysis', *Vehicular Communications*, vol. 14, Sep. 2018. DOI: 10.1016/j.vehcom.2018.09.004.

[2] M. Dibaei, X. Zheng, K. Jiang, S. Maric, R. Abbas, S. Liu, Y. Zhang, Y. Deng, S. Wen, J. Zhang, Y. Xiang and S. Yu, *An overview of attacks and defences on intelligent connected vehicles*, Jul. 2019.

[3] M. Wolf, A. Weimerskirch and T. Wollinger, 'State of the art: Embedding security in vehicles.', *EURASIP J. Emb. Sys.*, vol. 2007, Jan. 2007.

[4] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham and S. Savage, 'Experimental security analysis of a modern automobile', in *2010 IEEE Symposium on Security and Privacy*, 2010, pp. 447–462.

[5] G. A., *Jeep hackers are back to prove car hacking can get much worse*, 2016. [Online]. Available: https://www.wired.com/2016/08/jeep-hackers-return-high-speed-steering-acceleration-hacks/.

[6] C. Patsakis, K. Dellios and M. Bouroche, 'Towards a distributed secure in-vehicle communication architecture for modern vehicles', *Computers & Security*, vol. 40, Jan. 2013. DOI: 10.1016/j.cose.2013.11.003.

[7] M. Cheah, S. Shaikh, J. Bryans and P. Wooderson, 'Building an automotive security assurance case using systematic security evaluations', *Computers & Security*, vol. 77, Apr. 2018. DOI: 10.1016/j.cose.2018.04.008.

[8] C. Blommendaal, *Information security risks for car manufacturers based on the in-vehicle network*, 2015. [Online]. Available: http://essay.utwente.nl/68169/.

[9] J. Liu, S. Zhang, W. Sun and Y. Shi, 'In-vehicle network attacks and countermeasures: Challenges and future directions', *IEEE Network*, vol. 31, no. 5, pp. 50–58, 2017.

[10] P. Kleberger, T. Olovsson and E. Jonsson, 'Security aspects of the in-vehicle network in the connected car', in *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011, pp. 528–533.

[11] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham and S. Savage, 'Experimental security analysis of a modern automobile', in *2010 IEEE Symposium on Security and Privacy*, 2010, pp. 447–462.

[12] *Bosch. osi layers in automotive networks*, 2013. [Online]. Available: http://www.ieee802.org/1/files/public/docs2013/%20new-tsn-diarra-osi-layers-in-automotive-networks-0313-v01.pdf.

[13] I. Studnia, V. Nicomette, E. Alata, Y. Deswarte, M. Kaaniche and Y. Laarouchi, 'A survey of security threats and protection mechanisms in embedded automotive networks', Jun. 2013, pp. 1–12. DOI: 10.1109/DSNW.2013.6615528.

[14] K. B. Kelarestaghi, M. Foruhandeh, K. Heaslip and R. Gerdes, 'Vehicle security: Risk assessment in transportation', Apr. 2018.

[15] H. Guan, W. Chen, H. Li and J. Wang, 'Stride-based risk assessment for web application', *Applied Mechanics and Materials*, vol. 58-60, Jun. 2011. DOI: 10.4028/www.scientific.net/AMM.58-60.1323.

[16] M. Pratt, *What is an intrusion detection system? how an ids spots threats.* 2018. [Online]. Available: https://www.csoonline.com/article/3255632/what-is-an-intrusion-detection-system-how-an-ids-spots-threats.html.

[17] N. Shevchenko, T. A. Chick, P. O'Riordan, T. P. Scanlon and C. Woody, *Threat modeling: A summary of available methods*, 2018. [Online]. Available: https://apps.dtic.mil/sti/pdfs/AD1084024.pdf.

[18] M. N. Johnstone, *Threat modelling with stride and uml*, 2010. [Online]. Available: https://ro.ecu.edu.au/ism/88/.

[19] F. Shull, *Cyber threat modeling: An evaluation of three methods*, 2016. [Online]. Available: https://insights.sei.cmu.edu/sei_blog/2016/11/cyber-threat-modeling-an-evaluation-of-three-methods.html.

[20] S. van Winsen, *Threat modelling for future vehicles : On identifying and analysing threats for future autonomous and connected vehicles*, 2017. [Online]. Available: http://essay.utwente.nl/71792/.

[21] R. Wolthuis and F. Phillipson, 'Quantifying cyber security risks', in. Aug. 2019, pp. 20–26.

[22] J. Hughes and G. Cybenko, 'Quantitative metrics and risk assessment: The three tenets model of cybersecurity', *Technology Innovation Management Review*, vol. 3, pp. 15–24, Aug. 2013. DOI: 10.22215/timreview/712.

[23] S. Abraham and S. Nair, 'Predictive cyber-security analytics framework: A non-homogenous markov model for security quantification', *CoRR*, vol. abs/1501.01901, 2015. [Online]. Available: http://arxiv.org/abs/1501.01901.

[24] L. Moukahal and M. Zulkernine, 'Security vulnerability metrics for connected vehicles', in *2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, 2019, pp. 17–23.