The Need for Security in Distributed Automotive Systems

Stefan Seifert, Markus Kucera and Thomas Waas
Department of Computer Science, HS-Regensburg, Regensburg, Germany

Keywords: Automotive, Security.

Abstract: The papers main focus is on security in the automotive domain. It gives an overview about the current state of the art in this area. There is a trend to open today's vehicle architecture to technology known from the consumer segment (e.g. All IP Car). This is mainly motivated by cost reduction, reduced cabling effort and innovative functionality (e.g. car to car communication, intelligent navigation systems). By opening the architecture in such a way cars are getting more external interfaces which make them more accessible from the outside. Hence, an attacker does not need direct physical access to attack the car anymore but rather can use one of its wireless external interfaces. Using technology from the consumer segment does not only make the software and hardware development easier due to reusability but also makes the car an easier target. Therefore, additional research is needed to harden the automotive and make it more resistant.

1 INTRODUCTION

In the automotive industry, vehicle safety, and safety of the electronic car systems has been a major topic since many years. However, the security of such automotive system has not been of any great concern yet.

Nowadays modern automobiles have up to 100 different electronic control units (ECU) (Audi, 2009) and many different external interfaces (e.g. WLAN, Bluetooth, Mobile Broadband…). Considering the amount of ECU's and the different bus systems that are interconnected (CAN, MOST, LIN, FlexRay), the overall complexity of the system has increased and tends to increase even more.

Current development integrates Ethernet into automobiles (BMW Group, 2009) and replaces applications which till now were located in the car with applications positioned in the internet or the cloud, e.g. Navigation systems. Therefore, future cars will not only have integrated Ethernet as a bus system but might also be heavily interconnected with services on the internet.

Since standardized technology like Ethernet is used, it is easy to access the car without having to use expensive or proprietary equipment.

A look at current manipulation activities in this area underlines the argumentation given above. Today car manipulation is already reality, even though the attacker's intentions are slightly different. Examples are chip-tuning or unlocking special features (e.g. watching videos while driving which should only be possible while the car is not moving). The authors of (Dittmann et al., 2011; Tuchscheerer et al., 2011) give an overview about possible modifications and their implications.

In the automotive domain, it is important to guarantee the required system safety. If a security incident happens, it is essential to guarantee that the system's safety is not affected. Therefore, in future cars security issues have to be dealt with during the development of the safety case. Besides the safety issues, security incidents can result in money loss and loss of reputation.

It is not only important to handle incidents correctly but also to have tamperproof logging in place to gather forensic evidence about the incident so the circumstances can be analysed and reproduced afterwards.

2 SECURITY VULNERABILITIES

During the last years different vulnerabilities of automotive systems have been discovered. These can be classified in local vulnerabilities where the attacker has to have physical access to the vehicle and remote vulnerabilities where physical access is not needed.

Seifert S., Kucera M. and Waas T..
The Need for Security in Distributed Automotive Systems.
DOI: 10.5220/0004341001040110
In *Proceedings of the 3rd International Conference on Pervasive Embedded Computing and Communication Systems* (PECCS-2013), pages 104-110
ISBN: 978-989-8565-43-3

2.1 Local Vulnerabilities

In the following a list of published local attacks is given. Most of them are done via the CAN-bus, e.g. the major communication bus in today's vehicles.

- While opening a car window, the CAN messages are recorded and saved. Afterwards replaying the previously recorded messages leads to opening of a car window. If the attacker sends these messages in a loop, the pressing of the close-button does not succeed. Another attack over the CAN-bus involved malicious code, which sniffs for the vehicle speed information. This code monitors the vehicle speed and opens the car window using the previously described replay attack as soon as the cars speed exceeds 200km/h. Both attacks were only simulated (Hoppe and Dittmann, 2007).
- Gateways are used to isolate different networks from each other and only allow specified messages to be transmitted between the sub networks. In normal operation the On-Board Diagnosis Interface (OBD-II interface) is isolated from the internal CAN communication, which means no CAN-message can be transferred to the OBD-II interface. Therefore, the internal communication cannot be observed. Only when connecting a diagnostic device to the OBD-II port a connection to a target ECU is established via CAN and messages, which belong to this particular session, are transmitted. At the beginning of a session, the ECU and the device connected via OBD-II agree upon a CAN-ID that will be used during this diagnostic session. A bug in the gateway of a not disclosed car model leads to an eavesdropping attack. An attacker can set the CAN-ID to an ID that is already used in the internal network. This triggers a bug, which forwards all internal messages with the specified CAN-ID to the OBD-II interface (Hoppe et al., 2009), enabling unauthorized monitoring and writing of in vehicle information.
- Any component which has access to the CAN network can set the warning lights on or off. If the system is sending "on-messages", for example if the hazard lights are turned on, and an attacker immediately sends "off- messages" the hazard lights will stay dark. This makes a denial of service attack possible which floods the can bus with "off-messages". The threat agent for this scenario might be a thief who tries to steal a car. Even though, the anti-theft alarm system might detect an attack the horn would stay off and the hazard lights will stay dark (Hoppe et al., vol. 96, no. 1).
- Due to the violation of authenticity and integrity in CAN networks, it is possible to remove the airbag control system and replace it with a "fake" system (Hoppe and Dittmann, 2008). This fake system emulates the behavior of the air-bag system and therefore cannot be detected without additional means. This can be accomplished by simply sniffing the communication during a regular diagnostic session and then replaying the messages in the bus system.
- The authors of (Koscher et al., 2010) give a broad analysis of the automotive attack surface. They developed the program "CarShark" to sniff and inject packets into the cars CAN bus via the OBD-II interface. By sniffing the traffic, they were able to reverse engineer the protocol and take control of many ECUs. Besides sniffing, they also used fuzzing to discover new functions. With that information it was possible to reverse engineer the firmware of different ECUs. All those techniques lead to a number of possibilities to interfere with the normal operation of the car. The found vulnerabilities were tested on the road and it was possible e.g. to instantly lock the brakes or to permanently release the brakes so that the driver was not able to use them anymore.
- Also the media player – used by many audio players in today's car – can be misused for an attack. In (Checkoway et al., 2011) the firmware of a media player had been reverse engineered. The authors discovered a bug in the WMA parser, which leads to an exploitable buffer overflow. It was possible to create a special crafted wma file, which uses the vulnerability to execute malicious code. While playing the file in the cars audio player, it is possible to inject CAN messages into the system. Playing this wma file on a regular PC does not cause any side effects there, which makes it even harder to detect.

All described research clearly shows that there is a rather larger diversity in possible attack vectors. As already stated, the current vehicle architecture was developed with main focus on functionality and safety. For the future automobile which is expected to use Ethernet as one major network technology it has to be ensured that security is taken into account from the very beginning.

2.2 Remote Vulnerabilities

In the following a list of published remote vulnerabilities is given.

- A lot of research has been done in the area of keyless entry systems. Cryptographic attacks where the algorithm itself is being attacked are discussed in (Courtois et al.; Indesteege et al., 2008; Paar et al.). As well as relay attacks like (Francillon et al., 2010) where a sender and receiver is being used to transmit the car key's signal to the car. With this setup, it is possible to open and start a car from a distance without actually stealing the key. The theoretical distance from the sender to the receiver can be up to 3000km.
- Another attack vector described in (Barisani and Bianco, 2007) uses the Radio Data Systems Traffic Message Channels (RDS-TMC) capabilities of the head unit (or any other RDS-TMC devices like satellite navigation systems) to inject false RDS-TMC messages. The hardware used by the authors to accomplish this attack was GNU Radio. It is possible to create false messages that announce bad weather, full car parks, closed roads, security messages (which warn from terrorist incidents) etc. Those messages would lead to a recalculation of the route in the attacked car's navigation system, and therefore give an attacker free roads or free car parks.
- Attacks via tire pressure monitoring systems: Today, there are two different solutions for tire pressure monitoring systems. One solution monitors the rotation speed of the tires via the ABS system to recognize a pressure drop in the tire. The other solution uses additional sensors to measure the actual pressure of each tire. Those sensor are wireless and send the information to an ECU. Three problems have been analyzed with that kind of tire pressure monitor system (TPMS) (Rouf et al., 2010).
    - The communication between sensor and respective ECU lacks authorization, authentication and cryptography mechanisms. Consequently, it is easy to sniff the traffic between the sensor nodes and inject packets to trigger the low-pressure warning lamp.
    - The second problem that has been found is related to an implementation error in the TPMS ECU. After excessive experiments involving packet spoofing, the pressure monitoring ECU crashed and even a hard reset was not able to get it back into an operational state.
    - The third problem was privacy related, due to a unique identifier contained in every message. These IDs do not change during the lifetime of the car, which makes it very easy for a third party to track the vehicle.
- Bluetooth capabilities are found in many of today's cars. For example, mobile phones can be connected to offer hands free calling. Checkoway et al., (2011) experimentally validated two Bluetooth vulnerabilities in a recent car model.
    - An indirect attack requiring an already paired device, where the smart phone has to be compromised by an attacker. Afterwards, a buffer overflow in the Bluetooth protocol stack can be used to compromise the telematics unit of the car and deliver a malicious payload containing a Trojan.
    - Because it might be hard for an attacker to initially pair his Bluetooth device with a car the authors describe a direct attack where no pairing is needed. The only prerequisite is that some device has to be already paired so through sniffing the Bluetooth MAC address of the car can be discovered. This address is then used for pairing requests and in combination with a bruteforce attack the PIN can be cracked. This attack does not need any driver interaction and there is no warning displayed.

All the attacks presented so far require the attacker to be nearby the car. But with the increased use of cellular network modules in vehicles it is possible to create attacks where the attacker can be further away.

- In addition, to the Bluetooth attack the authors of (Checkoway et al., 2011) also found exploitable vulnerabilities in the cellular network module. The telematics unit was equipped with cellular network capabilities for location based services and automatic crash notifications, in which they found the following three vulnerabilities:
    - A buffer overflow in the AqLink (Airbiquity Inc, 2007) software modem, which transfers data via the voice channel.
    - The challenge response authentication method with a pre-shared key. The random number generator is always reinitialized to the same value whenever the telematics unit starts. Therefore, with sniffing, recording

the correct response makes a replay attack possible.
    - A bug in the code parsing during authentication was discovered which accepted incorrect but "carefully" formatted messages (To exploit this vulnerability an average of 128 calls is needed).

  In the end, it was possible to install a Trojan on the telematics unit, which allows the attacker to track the vehicle with GPS, send CAN message and also listen to conversations in the car.
- The term 'War Texting' was introduced by Don Bailey (Don Bailey, 2011). He discovered a possibility to unlock cars by just sending a SMS to it. This was possible because the car used an undisclosed product that allows remotely controlling the car. He used reverse engineering to discover the flaw.

3 THE MODERN AUTOMOBILE AND ITS SECURITY

In the past, cars were fully mechanical. From then until now, more and more electronic systems moved into the car – and even more will. Almost every functionality is controlled by ECUs, e.g. the safety features like the airbag and also the comfort features from the air conditioning system, electronic windows opener to the entertainment system.

Also the engine is controlled by an ECU, which improves the fuel usage and reduces emissions. Special equipment is needed for vehicle diagnostics and maintenance. Via the diagnostic interface (OBD-II) technicians are able to read the error codes of the different ECUs and can also configure these ECUs.

Two vehicle systems are not completely replaced by electric systems yet, e.g. the steering and braking system (even though, they are assisted by electronic systems). "X-by-wire" aims to replace every traditional mechanical system in the car by a purely electronic system. In such a car every major vehicle function will be controlled by ECUs. For example Nissan plans to introduce a steer-by-wire system by 2013 (Lavrinc, 2012).

Modern cars already have a broad spectrum of different interfaces such as WLAN, UMTs (e.g. for telematics services), NFC or Bluetooth etc. Furthermore, current efforts are made to integrate Ethernet into the car, with the future goal of an all IP based car (Glass et al., 2010).

This evolution can be described in three steps. In the beginning or the first step, there was a purely mechanical car. In the second step, the car was equipped with electronic systems but it is not interconnected with the environment. Now in the final step the car is equipped with electronic systems, which actually could be remotely controlled. Especially, due to the integration of external interfaces (WLAN, Cellular networks), an attacker might accomplish it.

It is also planned to equip cars with an app market (Continental, 2012; The Telegraph, 2012), where the concept remains the same as with nowadays mobile devices. Integrating this business model into cars opens many new aftermarket opportunities for the industry. It would be possible to extend the infotainment systems functionality. For example, the user could retrofit a navigation system with custom tailored apps. However, together with this new business model there comes also a security risk. All of today's mobile devices are vulnerable to jail breaking or rooting. Therefore, there is a high risk for this kind of attack to be feasible on future cars, too. Especially with in-vehicle-infotainment systems like AutoLinQ (Continental Automotive, 2012) or Meego (MeeGo, 2012) which are based on the Android OS. It is important to learn from mobile devices security problems, on both, application level as well as on operating system level, to avoid these problems on tomorrows vehicles (Schaub et al., 2011; Asaj et al., 2011).

For the safety of a car, security is required. The separation of security and safety is often not possible. To guarantee the safety, the system needs protection against malicious manipulation or attacks. As highlighted in (Frank and Spindler, 2011; Scheibert and Steurich, 2011) modern automotive microcontroller units (MCUs) already have different security features integrated (like trusted boot) which might be useful to accomplish this task.

4 FUTURE PROSPECT

In the previous chapters we highlighted the security problems of current and future cars. In this chapter we would like to give a perspective to possible solutions and open problems.

Not a possible solution but rather a technique that is often suggested is encryption or cryptography. Currently the inter-ECU communication is not encrypted. With Ethernet, anyone that has access to the network could eavesdrop on the communication channel. Therefore, encryption sounds like a logical step to take. This also applies to communica-

tion with the manufactures backend, where e.g. in a centralized approach the route for the navigation system is computed and sent to the car. But encryption has a rather large performance impact and the manufactures want the ECUs in the car to be cheap. SEIS (SEIS, 2012; Dr. bless, 2012) suggested to use IPsec to secure inter-ECU (transport mode) and backend (tunnel mode) communication. However, it has to be evaluated if IPSec or e.g. TLS are really suitable for the automotive environment. Still, encryption can only be one part of how to solve the problem, as shown in (Georgiev, 2012) the best cryptography is worthless if the implementation is faulty or it is wrongly used.

As already mentioned, the current automotive system architecture is not designed regarding security. Therefore, it is inevitable to overthink the system architecture with focus on security. Just as it is known from regular IT-networks, it might be necessary to separate automotive components in regard to their criticality in each sub network and restrict access to it. SEIS (Dr. bless, 2012) is proposing a similar approach, e.g. a three zone architecture model. But it is hard to identify all components which have the same criticality while considering all side effects. For example the GPS antenna or the distance sensor can be seen as having a low criticality. However, if highly critical components have access to them they could be a target for a safety relevant security attack.

Not only is it essential to make sure that design and implementation is secure, but also the security during the whole lifecycle of the car is very important. A car that was state of the art at the beginning of its production might not have this property the whole lifespan. In fact, finding vulnerabilities soon after the car's "release" to the market is quite possible, as it is often the case with software products in the desktop and server domain. Therefore, it is important to ensure safe, secure and in-time updates over the whole lifecycle.

Not only the transmission of the updates has to be reliable and secure but also the update process within the ECUs has to be error free and needs to take place without driver intervention.

Another possible attack vector is the backend, which provides updates or applications through e.g. an application store. Securing the backend, and its communication, has thus to be a major topic in order to prevent attackers to infiltrate a large number of cars. This poses some challenges because repair shops might need access to the backend to run vehicle diagnostics, which offers yet another interface to be targeted by attackers as well.

Finally the ECUs operating system has to be hardened against attacks using state of the art technology e.g. ASLR, DEP or mandatory access control.

5 CONCLUSIONS

Security in distributed automotive systems offers still a lot of unsolved challenges for the next years.

To create a secure automobile architecture a holistic view on the problem has to be created. Security cannot be fixed by just adding encryption or changing the architecture. A lot of different changes have to be done in order to create an overall secure system. One example is the development process. It has to be enhanced in order to not only consider system safety, but also system security. Therefore, security has to be tightly integrated into the development process from the beginning, when the automobile is designed during its manufacturing and until it is finally scrapped. This is not only a challenge for the manufactures but also for all other companies involved in the creation of the car. Since many subsystems are bought from suppliers, future vehicle integration has to focus not only on correct functionality and safety but also on the security of each subsystem and the overall system architecture.

REFERENCES

Audi, "Selbststudienprogramm 459: Audi A8'10 Bordnetz und Vernetzung," 2009.

BMW Group, Der neue BMW 7er: Entwicklung und Technik, 1st ed. Wiesbaden: Vieweg + Teubner, 2009.

J. Dittmann, T. Hoppe, S. Kiltz, and S. Tuchscheerer, Elektronische Manipulation von Fahrzeug- und Infrastruktursystemen: Gefährdungspotential für die Straßenverkehrssicherheit. Bremerhaven: Wirtschaftsverl. NW, Verl. für Neue Wiss, 2011.

S. Tuchscheerer, T. Hoppe, H. Adamczyk, M. Pukall, and J. Dittmann, "Herausforderungen an die Absicherung von IT Systemen in der Entwicklung, Betrieb und Wartung von Fahrzeugen," in Forschung und Innovation: 10. Magdeburger Maschinenbau-Tage ; 27. - 29. September 2011, Magdeburg: Univ, 2011.

T. Hoppe and J. Dittmann, "Sniffing/replay attacks on CAN buses: A simulated attack on the electric window lift classified using an adapted CERT taxonomy," (eng), 2nd Workshop on Embedded Systems Security (WESS' 2007), 2007.

T. Hoppe, S. Kiltz, and J. Dittmann, "Automotive IT-security as a challenge: Basic attacks from the black box perspective on the example of privacy threats," in Lecture notes in computer science; vol. 5775,

Computer safety, reliability, and security, Berlin [u.a.]: Springer, 2009, pp. 145–158.

T. Hoppe, S. Kiltz, and J. Dittmann, “Security threats to automotive CAN networks: practical examples and selected short-term countermeasures,” (eng), Reliability engineering & system safety, vol. 96, no. 1, pp. 11–25,

Tobias Hoppe and Jana Dittmann, “Vortäuschen von Komponentenfunktionalität im Automobil: Safety- und Komfort-Implikationen durch Security-Verletzungen am Beispiel des Airbags,” in Sicherheit, 2008, pp. 341-353.

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 Security and Privacy (SP), 2010 IEEE Symposium on, 2010, pp. 447–462.

S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, “Comprehensive experimental analyses of automotive attack surfaces,” in Proceedings of the 20th USENIX conference on Security, Berkeley, CA, 2011, pp. 6-6.

J. Hubaux, S. Capkun, and Jun Luo, “The security and privacy of smart vehicles,” IEEE Secur. Privacy Mag, vol. 2, no. 3, pp. 49–55.

U. E. Larson and D. K. Nilsson, “Securing vehicles against cyber attacks,” in Proceedings of the 4th annual workshop on Cyber security and information intelligence research, New York, NY, USA: ACM, 2008.

S. Pathak and U. Shrawankar, “Secured Communication in Real Time VANET,” in Emerging Trends in Engineering and Technology (ICETET), 2009 2nd International Conference on, 2009, pp. 1151–1155.

D. Eckhoff, C. Sommer, T. Gansen, R. German, and F. Dressler, “Strong and affordable location privacy in VANETs: Identity diffusion using time-slots and swapping,” pp. 174–181.

C. Lin and R. Shakya, “VANET worm spreading from traffic modeling,” in Radio and Wireless Symposium (RWS), 2010 IEEE, 2010, pp. 669–672.

N. T. Courtois, G. V. Bard, and D. Wagner, “Algebraic and Slide Attacks on KeeLoq,” pp. 97–115.

S. Indesteege, N. Keller, O. Dunkelman, E. Biham, and B. Preneel, “A Practical Attack on Keeloq,” IN EUROCRYPT, pp. 1–18, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.190.7835, 2008.

C. Paar, T. Eisenbarth, M. Kasper, T. Kasper, and A. Moradi, “KeeLoq and Side-Channel Analysis-Evolution of an Attack,” pp. 65–69.

A. Francillon, B. Danev, and S. Capkun, “Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars,” IACR Cryptology ePrint Archive, vol. 2010, p. 332.

A. Barisani and D. Bianco, “Hijacking RDS-TMC Traffic Information signals,” BlackHat, Las Vegas USA, 1-2 Agust 2007, http://www.phrack.org/issues.html?issue=64&id=5#article, 2007.

I. Rouf, R. Miller, H. Mustafa, T. Taylor, S. Oh, W. Xu, M. Gruteser, W. Trappe, and I. Seskar, “Security and Privacy Vulnerabilities of In-Car Wireless Networks: A Tire Pressure Monitoring System Case Study,” in Proceedings of the 19th USENIX Security Symposium, 2010.

Airbiquity Inc, “Whitepaper: aqLink® Overview,” http://www.m2mpremier.com/uploadFiles/aqLink_Overview.pdf, 2007.

A. Don Bailey, War texting: Weaponizing Machine 2 Machine. Available: http://www.isecpartners.com/storage/docs/presentations/isec_bh2011_war_texting.pdf (2011, Nov. 28).

D. Lavrinc, Nissan's Steer-by-Wire System Brings Us Closer to Autonomous Cars | Autopia | Wired.com. Available: http://www.wired.com/autopia/2012/10/nissan-steer-by-wire/ (2012, Nov. 02).

M. Glass, D. Herrscher, H. Meier, M. Piastowski, and P. Schoo, „SEIS“ – SICHERHEIT IN EINGEBETTETEN IP- BASIERTEN SYSTEMEN. ATZ elektronic, 2010.

Continental, With Continental the Apps Conquer the Road. Available: http://www.conti-online.com/generator/www/com/en/continental/pressportal/themes/press_releases/3_automotive_group/interior/press_releases/pr_2010_02_23_cebit2010_autolinq_en.html (2012, Feb. 07).

The Telegraph, Relaunched Audi A2 to include 'app' style customisations. Available: http://www.telegraph.co.uk/motoring/car-manufacturers/audi/7788442/Relaunched-Audi-A2-to-include-app-style-customisations.html (2012, Feb. 07).

Continental Automotive, AutoLinQ™. Available: http://www.autolinq.de/en/ (2012, Feb. 07).

MeeGo, In-Vehicle. Available: https://meego.com/devices/in-vehicle (2012, Feb. 07).

F. Schaub, B. Könings, and M. Weber, “Learning from Android,” in Automotive security: 27. VDI/VW-Gemeinschaftstagung, Berlin, 11. und 12. Oktober 2011, Düsseldorf: VDI-Verl, 2011.

N. Asaj, A. Held, and S. Schlott, “"Apps" im Fahrzeug - Ansätze und deren Sicherheits- und Privacy-Implikationen,” in Automotive security: 27. VDI/VW-Gemeinschaftstagung, Berlin, 11. und 12. Oktober 2011, Düsseldorf: VDI-Verl, 2011.

J. Frank and P. Spindler, “Security vs. Safety,” in Automotive security: 27. VDI/VW-Gemeinschaftstagung, Berlin, 11. und 12. Oktober 2011, Düsseldorf: VDI-Verl, 2011.

K. Scheibert and B. Steurich, “Sichere Mikroprozessorarchitekturen: Lösungsansätze aus der Halbleiterindustrie,” in Automotive security: 27. VDI/VW-Gemeinschaftstagung, Berlin, 11. und 12. Oktober 2011, Düsseldorf: VDI-Verl, 2011.

SEIS - Sicherheit in Eingebetteten IP-basierten Systemen — eNOVA - Strategiekreis Elektromobilität. Available: http://strategiekreis-elektromobilitaet.de/public/projekte/seis (2012, Nov. 03).

R. Dr. bless, C. Haas, and C. Werle, Eine sichere IPv6-basierte Architektur für Fahrzeugkommunikation. Available: http://strategiekreis-elektromobilitaet.de/public/projekte/seis/das-sichere-ip-basierte-fahrzeugbordnetz/pdfs/TP4_Vortrag1.pdf (2012, Nov. 03).

Martin Georgiev, Subodh Iyengar, Suman Jana, Rishita Anubhai, Dan Boneh, and Vitaly Shmatikov, "The most dangerous code in the world: validating SSL certificates in non-browser software," in ACM Conference on Computer and Communications Security, 2012, pp. 38–49.