FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

ASSIGNMENT OF MASTER'S THESIS

Title:	Tesla Model 3 Internal Network Security Analysis
Student:	Bc. Filip Machala
Supervisor:	Ing. Jiří Dostál, Ph.D.
Study Programme:	Informatics
Study Branch:	Computer Security
Department:	Department of Information Security
Validity:	Until the end of winter semester 2021/22

Instructions

Tesla Model 3 is a brand new generation electric car with a new architecture and infrastructure of control units. Unlike the cars from classic carmakers, it doesn't use CAN bus but Ethernet based networks for most of the communication instead. Get familiar with cybersecurity issues in the automotive industry. On a testing car perform security analysis of an internal network that connects electronic units (ECU). The main target is the connection between the autopilot unit (ACU) and the multimedia unit (MCU). Describe network topology and used protocols. Map attack surface and create a threat model. Test chosen vulnerabilities and create your exploits in case of need. In case of discovering "zero day" exploits start process of responsible disclosure. Document everything, evaluate security impact and suggest remediation.

References

Will be provided by the supervisor.

prof. Ing. Róbert Lórencz, CSc.
Head of Department

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
Dean

Prague May 25, 2020

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Master's thesis

Tesla Model 3 Internal Network Security Analysis

Bc. Filip Machala

Department of Information Security
Supervisor: Ing. Jiří Dostál, Ph.D.

May 28, 2020

Acknowledgements

I want to thank my supervisor Ing. Jiří Dostál, Ph.D., for all his valuable advice and for the time he spent helping me with this thesis. Also, I am grateful to my friends for their comments and words of support.

I am very grateful to my parents and family for supporting me throughout my studies.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as a school work under the provisions of Article 60 (1) of the Act.

In Prague on May 28, 2020 . . . . . .

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Machala, Filip. *Tesla Model 3 Internal Network Security Analysis*. Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2020.

Abstrakt

Tato práce se zabývá bezpečností auta Tesla Model 3. Hlavním cílem práce bylo analyzovat zabezpečení interní sítě založené na Ethernetovém protokolu. Analýza vychází z upravené verze směrnice PTES a zahrnuje vytvoření modelu hrozby, který identifikuje zranitelná místa. Následuje jejich hloubková analýza. Analýza ukázala, že interní síť automobilu je chráněna před vnějšími hrozbami, avšak komunikace v místní síti není zabezpečena. Všechna zjištění jsou shrnuta v závěru práce.

Klíčová slova Ethernet, bezpečnostní analýza, Tesla Model 3, Automotive Ethernet, BroadR-Reach

Abstract

This thesis is about security of the Tesla Model 3. Main goal of the thesis was to analyze security of Ethernet based internal network. Analysis follows modified version of the PTES guideline. It includes creation of the threat model that identifies vulnerabilities followed by their in depth analysis. Analysis has showed that car's internal network is protected against external threats, however communication on local network is not secured. All findings are summarized in the conclusion of the thesis.

Keywords Ethernet, security analysis, Tesla Model 3, Automotive Ethernet, BroadR-Reach

Contents

List of Figures

List of Tables

Introduction

Computer security has become an important part in today's life as still bigger portion of it is moving to online sphere. These days, most people know to some extent, that their data could be stolen and abused and that they need to protect them. They are already quite well used to being careful when using a computer or a mobile phone. But in recent years technology has found its way also in a lot of other things that we use on the daily basis. One of them is a car, which has become capable of autonomous drive and nowadays it contains far more computing power than people actually think. And when you keep in mind, that cars are also equipped with various connection possibilities (LTE network, Wi-fi, physical ports), the focus on the cybersecurity in cars starts to be very important. Automotive industry is trying to react to this request. One of the outcomes is, that manufacturers started to use technologies not typical for their field. One of this case is Tesla using Ethernet based network in its Tesla Model 3. Cars used to use only CAN bus networks. Usage of Ethernet based network is an innovation in the industry. But how does this affect security of the car?

Finding an answer to this question is the aim of this thesis. Main goal of this thesis is to analyze security of the internal network in the Tesla Model 3 with main focus on the Ethernet based network.

At first this thesis makes reader familiar with information about automotive industry and its specifics. After that past attacks and information about the car are discussed. Next, threat model identifying possible vulnerabilities is constructed. Based on threat model, vulnerabilities are in-depth described. The outcome of this process analysis is a summary of the findings. Chosen attacks are demonstrated in the end of the thesis .

Chapter 1

Goal proposal

1.1 Evolution of the cars

Cars have become computers on wheels during the last decade. They have learned to drive autonomously and their ability to do so has been radically improving during recent years. From ability to perform an emergency braking to prevent an accident, up to cars capable of fully autonomous drive improving both passenger safety and comfort. With all the benefits these technology implementations have also comes the request for higher security level of the system in the cars. For example, modern car infotainment system (including personal assistant, GPS navigation, web browser, ability to pair with a phone etc.) provides completely new user experience that is miles away from cars manufactured a decade ago. But this rather extensive system can also gather and store quite a lot of personal data that we - the users willingly or even unwittingly provide by using these systems. And if the system is not well protected, these data can be stolen. Also all systems controlling the car (various drive and anti-collision assistants, automatic opening and closing of the doors and windows, air conditioning...) can potentially be abused if they are not enough protected against unauthorized access. And the consequences can be fatal. By stating so, there comes the obvious question: Are modern cars really secure in terms of cybersecurity?

With many successfully executed cyber attacks on cars from different manufacturers, it seems like we know the answer. Seriousness of these attacks vary from being able to sniff diagnostic data through exploiting wireless unlocking of the car and being able to unlock it without original key fob to gaining full remote control of the vehicle while it is being used. All that can lead to threatening lives of the passengers. Considering all these information, question about cybersecurity is becoming more and more serious.

1.2 Tesla Model 3 internal network design

All car makers are trying to make their cars safe and secure. Their objective is not easy because they always work with limited budget to maximally fulfill the objective.

Tesla became leading manufacturer of electric cars in just two decades. Tesla was found in 2003 and is still young company without history in automotive industry in comparison with classic car makers with tens of years of experience. Despite lack of experience in comparison with these car makers Tesla sold the most electric cars in first quarter of 2020 [1]. Taken these numbers into considerations it is clear that people like Tesla's car. Its newest model called Tesla Model 3 is the most affordable model and at the same time it is coming with the newest computing architecture and advanced auto pilot functions. Tesla model 3 represents new generation of cars using Ethernet based networks for the most of the communication instead of the older CAN bus networks. While using Ethernet based networks eliminates vulnerabilities of the CAN bus networks it brings vulnerabilities from Ethernet based networks. Ethernet based networks are something new to automotive industry and their security has not been analyzed properly.

1.3 Goal of the thesis

As Ethernet based networks are new in the automotive industry, their security is not analyzed properly for various reasons. One reason is, that to do that, one needs a car with this type of network. Buying a car only for testing purposes is not affordable for the most enthusiasts and even majority of professionals. I was given a unique opportunity to work with a team on security analysis of the Tesla Model 3, which made the realization of this type of network analysis possible for me.

Main focus of this thesis is to analyze security of the Ethernet based internal network in the Tesla Model 3. Analysis must contain:

1. Identification of devices using Ethernet based networks.
2. Description of possible ways to connect to Ethernet based networks.
3. Threat model identifying possible vulnerabilities.
4. In depth analysis of these vulnerabilities.
5. Summary of found vulnerabilities.

1.4 Procedure of analysis

Execution of the analysis must be well thought to achieve convincing outcome.

1.4.1 Automotive industry specifics

As automotive industry has its own specialties, my first step must be getting familiar with it. Automotive industry uses Automotive Ethernet not classical Ethernet that is used in companies, schools and houses. Required functionalities differs from other industries [2]. Requirement on the lowest possible price is the same, but also many other requirements like reliability and weight are very important. People do not care if cable which is in the wall of their apartment weights 5 or 10 kilograms. On the other side, in a car every kilogram have impact on fuel efficiency and riding characteristics. Safety systems in cars are time critical what adds even more requirements. Not only maximum speed is important but also reliability that message will be delivered in given time. To fulfill all requirements of the industry Automotive Ethernet was created.

1.4.2 Using standards

To cover everything properly a methodology is needed. There is no standard for executing security analysis in the automotive industry. However many standards define process of penetration testing which has many similarities. Slightly modifying some of these standards should be a good start.

1.4.3 Research

Learning from others is very important. Research for successful attacks executed in the past should definitely help to prepare. Also searching for any information related to this topic is absolutely essential.

1.4.4 Analyzing vulnerabilities

Using prepared methodology helps with practical part. Creating realistic threat model and after that analyzing identified vulnerabilities is main part of this thesis.

Chapter 2

Theory

This chapter describes important terms when talking about cybersecurity in the automotive industry. Difference between safety and security and the most common terms are discussed to make reader familiar with terms used throughout whole thesis. Tesla's responsible disclosure guideline is described. In the end standards used for penetration testing are discussed and guideline used for this analysis created by modifying PTES guideline is described.

2.1 Basic terms

Definitions of basic terms used in the thesis.

Penetration test Penetration test is an authorized simulated attack on computer system. The goal is to obtain unauthorized access into the system. The final outcome is a pentest report covering all identified vulnerabilities [3].

Ethical hacking Ethical hacking uses various methods and activities to test the subject. Ethical hacker has wide and deep knowledge from various fields. Penetration test is one of these activities [3].

Attack vector The path or method that enables malicious activity or discloses vulnerabilities [3].

Attack surface Set of attack vectors [3].

Threat A potential danger that may lead to vulnerability and security breach. Threats exist everywhere. We need threat management so they don't lead to vulnerabilities [3].

Vulnerability It is a weakness that can be used for security breach [3].

Exposure

Exploit Software, payload or data that exploits the security vulnerability. Result is an unintended behavior or security breach [3].

Zero-day vulnerability A Zero-day vulnerability is a vulnerability newly found, therefore it is still unknown to vendors. It is the most dangerous vulnerability. From the moment that Zero-day vulnerability is published, related system is no longer secure [3].

Payload Transmitted data include headers information and meta-data. Payload refers to actual data in the transmitted data [3].

Brute force attack Form of attack that is often used to crack passwords. It is one of the simplest attacks. It uses all possible combinations of the given characters to create all possible passwords. Downside of this attack is the time complexity needed to succeed [4].

Buffer overflow This condition exists when a program attempts to put more data in a buffer that it can hold or when the program tries to save data in a memory past a buffer. Here can be something more [5].

Backdoor Backdoor is an undocumented way of gaining access to a program, service or even entire system. Backdoor access bypass normal authentication mechanics [3].

Security and safety Terms security and safety are often misinterpreted. Term of security is used to refer to the protection of individuals, organizations and assets against external threads. These can be criminals, weather, animals and so on. "Security is the safeguard that ensures our safety remains constant." For example if the company has servers, it wants to protect them from thieves but also weather conditions to ensure they won't break. Safety refers to keeping us protected from things that can cause harm or damage. This requires our participation. For example, we can't affect weather So when the road is icy we need to slow down to keep us safe [6].

Security by obscurity Security by obscurity refers to a process of hiding something and describe it as secure instead of securing it by proper way for example by authentication or encryption [7].

Controller Area Network (CAN) network Controller Area Network is serial communications bus used in in-vehicle communication. It was developed by Robert Bosch GmbH. It provides simple, robust and efficient communication [8].

Automotive Ethernet Automotive Ethernet is a physical network that is used to connect components within a car using a wired network. It is designed to meet the needs of the automotive market, including meeting electrical requirements (EMI/RFI emissions and susceptibility), bandwidth requirements, latency requirements, synchronization, and network management requirements. To fully meet the automotive requirements, multiple new specifications and revisions to specification are being done in the IEEE 802.3 and 802.1 groups [2].

BroadR-Reach BroadR-Reach is Broadcom's 100Mbps PHY implementation that uses technologies from 1G Ethernet to enable 100 Mbps transmition over a single pair in both directions [2].

2.2 Used commands

Nmap [9] Nmap (Network Mapper) is a free and open source utility for network exploration and security auditing. It was first released in 1997. From that time Nmap became world's most popular security scanner. It is very powerful and complex tool. It can detect available hosts on the network, services those hosts are offering, used operating systems and other characteristics.

ARP-scanner [10] ARP scanner is a free open source software for scanning the network. It uses ARP to discover any devices on the local network.

2.3 Tesla's Responsible Disclosure Guidelines

[11] Tesla has its own guideline for security researchers. Any found vulnerability must be reported to Tesla before publishing it to general public. Tesla will not take legal action against security researcher but he must follow the guideline which include:

1. Provide details of the vulnerability and information how to reproduce it.
2. Researchers could not modify or access data that does not belong to them.
3. Do not compromise the safety of the vehicle or expose others to an unsafe condition.

4. Security research is limited to the security mechanisms of the Infotainment binaries, Gateway binaries, Tesla-developed ECU's, and energy products.

2.4 Standards

There are many standards/methodologies for penetration testing [3]. Most used standards and frameworks are:

1. PTES (Penetration Test Execution Standard)
2. OWASP (Open Web Application Security Project)
3. NIST (National Institute of Standards and Technology)
4. ISSAF, OSSTMM and others. . .

All these standards are about performing penetration test. My security analysis is very similar to the penetration testing in some situations. I have chosen PTES [12] as template for my own guideline that I am using.

2.4.1 PTES guideline

I am using PTES [12] which describes guideline how to structure the penetration test. It does not describe actual execution of penetration test but include all its part to make it systematic. It has 7 sections:

1. Pre-engagement Interactions
2. Intelligence Gathering
3. Threat Modeling
4. Vulnerability Analysis
5. Exploitation
6. Post Exploitation
7. Reporting

2.4.1.1 Pre-engagement

Main focus of this section is to define scope of the penetration test. This is one of the most important parts of the test, where tester defines what should be tested and how. There is a big difference between intention to test 1000 IP addresses and get some basic information what services are available there, and testing just one IP address but with intention to discover all 1000 services that

are available there. To define the scope correctly, scoping meeting between the customer and ethical hacker should be arranged. They have to agree on scope of the penetration test, especially IP ranges, domains and if they should test also IP ranges and applications they get access to while testing. It is very important for the company which performs the test to get all information about third parties whose services can be encountered during the testing. This is very important because the company performing penetration test may potentially break law by testing services of the third party with whom it does not have legal agreement about testing. Other problem that can occur is called "scope creep". This means that testers are testing something that wasn't planned and therefore it was not included in the agreed price. It happens very often and it is quite a big problem for testing company, because they happen to do something they are not paid for. Scope meeting should prevent this from happening.

Both sides also have to come to an agreement on metric how to measure amount of work done. The most used metric are man-days. This can be easily converted into time that the tester will spend on the designated task. Another important thing that should not be overlooked is to define evidence handling. This is very important to ensure both sides know how to handle evidence.

2.4.1.2 Intelligence Gathering

This section is all about gathering as much information against a target as possible. Standard defines information gathering for penetration testing web applications which is slightly different from my case. I use this section of standard as template for mine method. I divide available information into two groups:

1. Information acquired from public sources.

2. Information acquired from car.

2.4.1.2.1 Information acquired from public sources Any information acquired from public sources without access to car belong to this category. Main source of information is internet and then other papers, thesis and books. Because of very fast developing and updating Tesla's software, the classic sources as papers and books are quickly outdated. Therefore internet is better source of information in this case. There are much more blogs and threads on different forums that are related to this topic. But all the information gathered from the internet have one common problem, they are not validated by anyone and everything acquired from this source needs to be validated. However, it still really helps to create clearer image about car and used principles.

2.4.1.2.2 Information acquired from car Information acquired by interacting with car belongs to this section. Based on the type of access one has to a car, I divided information based into 3 groups:

1. Without access to the car interior.
2. With the key card.
3. As a mechanic.

All information acquired in this section are credible, as they are directly from testing car, unlike the previous case 2.4.1.2.1. This division helps in next steps of analysis, especially in threat modeling section.

Without access to interior Here belongs everything one can find about the car from exterior of the car.

With te key card To this category belongs everything one can find about the car from interior of the car with key card. This represents normal owner of the car.

As mechanic This category involves any information acquired from the car including any disassembling needed to access parts that are hidden to normal user.

2.4.1.3 Threat Modeling

Threat modeling is the next part of the test. Cooperation with the client is essential in this process. Since PETS standard focuses on penetration testing network of the corporations, I need to modify it for my purpose.

It is a process of modeling attackers point of view and modeling potentiall impacts on the car. High level threat modeling process consists of 4 steps:

1. Gather relevant documentation
2. Identify and categorize primary and secondary assets
3. Identify and categorize threats and threat communities
4. Map threat communities against primary and secondary assets

2.4.1.3.1 Asset Analysis Standard defines this part as threat modeling exercise and asset-centric view on all assets and business processes supporting them, included in the scope. I am modifying this part to suit it for my case. I need to identify all assets that are important for the owner of the car.

2.4.1.3.2 Process Analysis All assets are part of some process. Businesses protect their assets, so they can use them and nobody else can. Therefore all processes that use these assets need to be secure. Identification of these processes is main purpose of process analysis.

2.4.1.3.3 Threat capability analysis After identifying possible threats we need to analyze them to build accurate threat model. I need to set score for each threat. This score consists of multiple parts:

1. Needed tools
2. Accessibility
3. Communication mechanics

Each part has score from 0 to 2.5, where 0 is the least IMPORTANT and 2.5 is the most IMPORTANT. Score of the threat can be from 0 to 10 after summing up all parts.

2.4.1.4 Vulnerability Analysis

"Vulnerability testing is the process of discovering flaws in systems and applications which can be leveraged by an attacker." Flaw can be anything from misconfiguration of the host or service to the bad security design of the application. Process of finding flaws differs depending on tested component. As the car uses classic Ethernet protocol there are many similarities with classic corporate networks.

Tester should properly scope depth and breadth to meet the goals and requirements.

2.4.1.4.1 Active testing "Active testing involves direct interaction with the component being tested for security vulnerabilities." This way can be tested low level components such as the TCP stack on a network device but also components such as web based application. Interaction with the target component can be done in two ways: automated and manual. Using automated tools is standard these days.

Network/General Vulnerability scanners "An automated port based scan is generally first step in a traditional penetration test." It gives a basic overview of what devices are on the target network. Port based scanning discovers which ports on the remote host are open and able to receive a connection. Port can be in one of the possible states:

1. Open - the port is able to receive data
2. Closed - the port is not able to receive data

Scanners can refer port as filtered. This happens when it cannot accurately determine whether a port is open or closed.

Service scanning After the open ports are known, services running on these ports should be identified. This is more complex process than the port scanning. Process called banner grabbing can be used in this purpose. This includes connecting to the specific port and examining returned data to identify used service or application. HERE CAN BE MORE ABOUT VPN AND OTHER

2.4.1.4.2 Passive testing Passive testing is done without any direct interaction with the target. Target does not know that we are testing it.

Metadata analysis Analyzing metadata instead of data directly can give attacker unwanted information that can be considered as a security flaw. Metadata are commonly used in documents on the internal network of the company but any leak of this metadata to public is unwanted.

Traffic monitoring Traffic monitoring is often used to gather information about target. Way to accomplish this is to listen on the local network and log whole communication for later analysis. Any unencrypted communication on the local network can be completely analyzed using this method.

2.4.1.4.3 Research "Once a vulnerability has been reported in a target system, it is necessary to determine the accuracy of the identification of the issue, and to research the potential of the vulnerability within the scope of the penetration test." Many THINGS can help us to verify an issue:

1. **Vulnerability Databases** should be used to verify the vulnerability on a target system.
2. **Vendor Advisories and change logs** can provide useful information about possible vulnerability.
3. **Exploit Databases and Framework Modules** are other sources for tester. There are more sites that are actively maintained therefore tester should use more than one site for better results.
4. CAN BE MORE HERE

2.4.1.5 Exploitation

"The exploitation phase of a penetration test focuses solely on establishing access to a system or resource by bypassing security restrictions."

2.4.1.5.1 Overall Objective In the pre-engagement interaction phase with the customer, a clear definition of the overall objectives of the penetration test should have been communicated. During the exploitation phase, the biggest challenge is to identify the path of least resistance into the organization. However, this breach must remain undetected and needs to have the biggest impact on the ability of organizations to generate revenue.

2.4.1.5.2 Countermeasures Countermeasures are preventive technologies or even methods whose purpose is to deny successful exploit of vulnerability. These can include various alarms that can be triggered. Ultimate purpose is to remain stealth without triggering any alarms. There are many types of countermeasures, some of them listed here:

1. Anti-Virus
2. Encoding
3. Packing
4. Encrypting

2.4.1.5.3 Exploit customization Exploits usually targets specific version of system. Penetration tester should be able to modify already published exploit in order to attack the system. It is often required to simulate the victims infrastructure to ensure, that exploit will work. Having a working infrastructure makes exploitation phase much easier.

2.4.1.5.4 Zero-Day Angle If none vulnerabilities were found Zero-Day angle if often the last resort for most penetration testers. This type of attack is trying to discover new vulnerabilities that are not publicly known yet. This type of attack is extremely difficult to execute and requires deep knowledge of the target. There is no defined way how to achieve the discovery of vulnerabilities. Here the creativity has no limits. Most common methods used in practice are shown bellow:

- **Fuzzing** is the ability to recreate a protocol or application and attempt to send data at the application in hopes of identification of a vulnerability. In the case of fuzzing, the attacker is trying to create vulnerability for specific version of software that has not been discovered yet.
- **Source code analysis** Looking into the source code, if a tester has the access to it, is a potentially good way to identify flaws within the application. Zero-day vulnerabilities can also be found using this method.

2.4.1.6 Post Exploitation

"The purpose of the Post-Exploitation phase is to determine the value of the machine compromised and to maintain control of the machine for later use." The value of the machine is determined by its ability to access valuable assets. This phase describes methods that should help the tester to identify and document sensitive data, configuration settings, communication channels and relationships with other network devices that can be used to gain further access to the network.

2.4.1.6.1 Rules of Engagement Rules of engagement specific to the post-exploitation phase should ensure that client's systems are not exposed to unnecessary risk by actions of the tester and to ensure that both sides agree on procedure to follow during the post-exploitation phase of the test.

Most important rules are to protect the client and to protect the tester.

Protect the client Purpose of these rules is to ensure that the client's data are not exposed to any risk. Tester can not modify services of the client's infrastructure for whatever purpose without agreement of the client. All actions taken by the tester must be documented in details. The tester must ensure, that all devices connected to the network and data stored on them are owned by the client. Any device or service that is used to maintain access to the client's systems must employ some kind of authentication, to deny creating any new vulnerabilities. All data gathered during the tests, stored on the tester's device must be encrypted. No logs can be modified from the client's systems without specific authorization by the client.

Protect the tester The tester must ensure that he is authorized to perform all actions that are necessary for complete test. This authorization must be included in the contract between the tester and the client. The tester must also ensure that he has authorization to services provided by 3rd party that user wants to test. Using full drive encryption for all systems that store and receive client's data. In all cases, it is the most important to obey laws and restrictions of current country. The tester must always know them, because they are far more important than restrictions in the contract.

2.4.1.6.2 Infrastructure analysis The network configuration of a compromised device can be used for further analysis of additional subnets, active network devices such as routers, critical servers, name servers and relationships between devices. Gathered information can be used to identify additional targets for future testing of the client's network.

2.4.1.6.3 Pillaging Pillaging refers to obtaining information (i.e. files containing personal information, credit card information, passwords, etc.)

from targeted hosts relevant to the goals defined in the pre-assessment phase. This information can be obtained to satisfy goals or as a part of the pivoting process to gain further access to the network. This data can be stored on different places, therefore basic knowledge of systems and programs is crucial. The tester should analyze multiple programs and services such as: database severs, name servers, deployment services, dynamic host configuration server and so on.

2.4.1.6.4 Persistence The next step after gaining access to the target is ensuring persistence for future use. There are many ways how to achieve this goal. The most common are:

- Installation of backdoor that requires authentication.
- Creation of alternate accounts with complex passwords.
- Installation and/or modification of services to connect back to the system. User and complex password should be used as a minimum; use of certificates or cryptographic keys is preferred where possible. (SSH, ncat, RDP). Reverse connections limited to a single IP may be used.

2.4.1.6.5 Cleanup The cleanup process describes the requirements for cleaning the system after the penetration testing has been finished.

- Remove all executable files from a compromised system, including all scripts and temporary files. It is preferred to use secure delete of all files.
- Return all system settings to its original value, if they were modified during the testing.
- Remove all backdoors and/or rootkits installed.
- Remove any user accounts created during the testing.

2.4.1.7 Reporting

Last phase is all about presenting results of the test. This can be done in many ways and standard describes one of them. It splits report into two major parts to communicate the objectives, methods and results of the testing.

First part is executive summary. This part presents specific goals of the penetration test and the high level findings. The intended audience are people in charge of the security program as well as all members of organization which may be impacted of the threats. This part describes overall purpose of the test. It explains overall risk score to simplify understanding for the audience. It provides general findings of the test with recommendation of the tasks to

resolve the risks. It can also contains strategic roadmap as prioritized plan for remediation of the insecure items found.

Second part is technical report. This part presents in depth details on the whole process of testing. All parts are detailed described and documented in this report. This thesis is form of technical report.

Chapter 3

Intelligence gathering

This chapter offers to reader gathered information about topic from various sources. Except classical scientific sources like books and papers also information from not scientific sources like public forums on the internet used by enthusiasts. Information from not scientific sources must be validated but provide nice vision of actual situation.

3.1 Information acquired from public sources

3.1.1 Known attacks in the past

Various attacks breaking security of the cars were already done in the past. These attacks can help me to better understand possible threats in the automotive industry.

Paper called *Lock It and Still Lose It – On the (In)Security of Automotive Remote Keyless Entry Systems* [13] revealed flaw of keyless entry systems that affects millions of vehicles. This flaw does not relate to the networks of the cars but demonstrates how huge effect can one flaw has.

In 2013 attack that takes control of the vehicle was revealed [14]. These attack revealed that connecting to internal network of the car can lead to gaining full control of the car. Reaction of the manufacturers was not expected. They did not considered connecting to the internal network of the car as real threat. On the other side they argued that accessing internal from inside of the car is no challenge.

In 2015 attack realized by Charlie Miller and Chris Valasek [15] revealed how important is security in the modern car. They created exploit that uses vulnerability in one of the units. This exploit enables them to connect to internal CAN network. After gaining access to CAN network they are able to affect critical functions of the car like steering, braking and accelerating. This threaten safety systems of the car that can put in danger lives of the passengers.

3.1.2 Owner's manual

Owner's manual is good source of information about the car [16].

Car uses Internet connection for many purposes. There are 2 options how car can connect to the internet. First is cellular network. This is pre-configured and user do not need to do anything to make this work, it will work automatically. Another way to connect to the internet is by using Wi-Fi. Wi-Fi connection is needed for most software updates to download.

3.1.3 Forums

Forums on the internet connects enthusiasts from all around the world. Their findings can be very helpful.

I found network diagram on the forum [17] 3.1. This diagram describes all CAN networks in the car. This gives me picture how everything is connected in the car. From this diagram ACU and ECU units looks really important. There is a project which purpose is to acquire and decode performance and

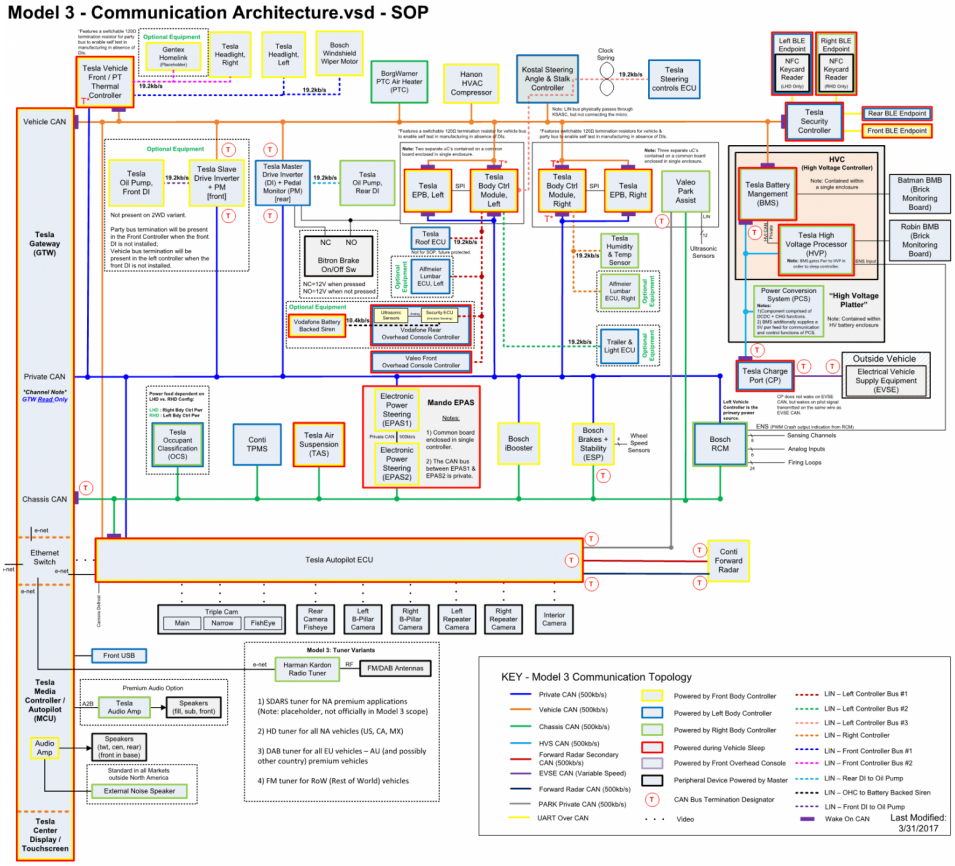

Figure 3.1: CAN diagram network[17]

other data from the CAN network in the Tesla Model 3. This project is still developing but has already achieved some success. They were able to trace CAN messages about motors power and battery power. They created application that shows these information in real time. All data are taken from diagnostic port located in the back of the central tunnel in the car.

Ethernet network Because Ethernet network is used, I need to find out its configuration. I was not able to found the exact network configuration of Tesla model 3, but I managed it for Tesla model S. It is older but more luxurious model. This source claims, that network uses private range 192.168.90.0 with network mask 255.255.255.0. Central console uses IP address 192.168.90.100 and then there are another two devices [18]. As manufacturers usually use similar solutions for different products to optimize costs, I suppose that same private range is used in the Tesla model 3. This premise should help me, but it needs to be confirmed.

3.1.4 CAN network

CAN network [8] was publicly released in 1986 to provide simple network to reduce wiring and to allow multiple microcontrollers to communicate on a single bus. The CAN bus acts as a backbone for many systems in the car, and signals are multicast over the bus. That means, that every device on network can see all messages on the network. Safety critical vehicle information such as engine control, door locks, anti locking braking, is passed on the CAN bus. The CAN bus is a multi-master differential communication system. The messages are multi-cast, meaning every microcontroller and component connected to the CAN bus receives each message. Usually car has several CAN bus networks each for specific purpose. One for operation and communication of the powertrain other for multimedia system of the car and so on. Research shows that the typical CAN bus is extremely vulnerable to attacks. Vulnerabilities listed below were identified:

1. Multicast Messaging: When a message is sent to the CAN bus, it has no specific destination. Every access point or controller on the bus has access to all messages. Passive attackers could listen in on the communication with ease.

2. Lack of Authentication: The typical CAN bus has no authentication process. Nodes do not have a process in place to ensure the message they receive is from a valid source.

3. Lack of Addressing: Nodes typically have no identification address, which allows all (real or malicious) nodes to send or receive information without any verification that the source of information is valid.

4. Common Point of Entry: Once an attacker has access to the CAN bus, there is no limit on type of parameters the attacker can obtain. Usually, one common gateway is used to connect all vehicle CAN bus systems.

5. Limited Bandwidth: The CAN bus has a limited bandwidth. This creates a challenge to allow any robust authentication process to be implemented.

6. Lack of Encryption: CAN bus is designed for ease of use. Data over the network is not encrypted making aftermarket ECU manipulation easy. To implement a robust encryption of data the CAN bus would need a larger bandwidth than what is available currently.

7. Multi-System Integration: Multiple suppliers integrate onto these CAN bus networks. Given that there is typically no security measures in the CAN bus standards, there is no objective by these system suppliers to create a secure communication protocol.

All these vulnerabilities can be exploited after gaining access to the CAN bus network by an attacker.

3.2 Information acquired from the car

3.2.1 Without access to the interior

The easiest information one can find without access to the interior are VIN number and license plate of the vehicle. VIN number is printed on the bottom of the windscreen. After using this number in online VIN database I am getting more information. This Tesla Model 3 was registered in 2019 and has two electric motors, what means it is AWD (all wheel drive).

Back mirrors can be disassembled without any problems. There are antennas in the back mirrors as shown in the pictures. There is Bluetooth and LTE antenna in the right back mirror 3.2 and Bluetooth antenna in the left back mirror 3.2.1.

3.2.2 With the key card

With the key card one can access interior of the car. Main input device is touchscreen, which has no connectors. There are only two USB connectors 3.4 in the front shelf. Then one can access charging connector of the car. It can be opened via the touchscreen.

There are NFC readers on the each side of the car between front and back doors. Same NFC reader is on the central console in the interior.

Figure 3.2: Right back mirror

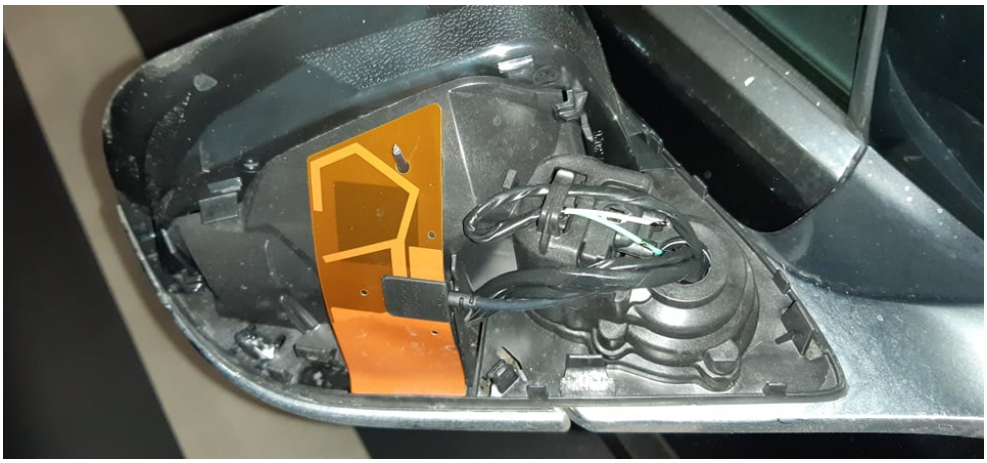

Figure 3.3: Left back mirror

3.2.3 As mechanic

3.2.3.0.1 Interior of the car As mechanic I can disassemble dashboard panels to access units directly. I want to be able to access the MCU and the ACU unit.

Physical location of units in the car

MCU and ACU unit After disassembling front desk I have access to MCU unit directly 3.5. The ACU unit is directly behind it. They share same cooler and therefore are assembled together. There is really limited access to the

Figure 3.4: USB connectors in the front shelf

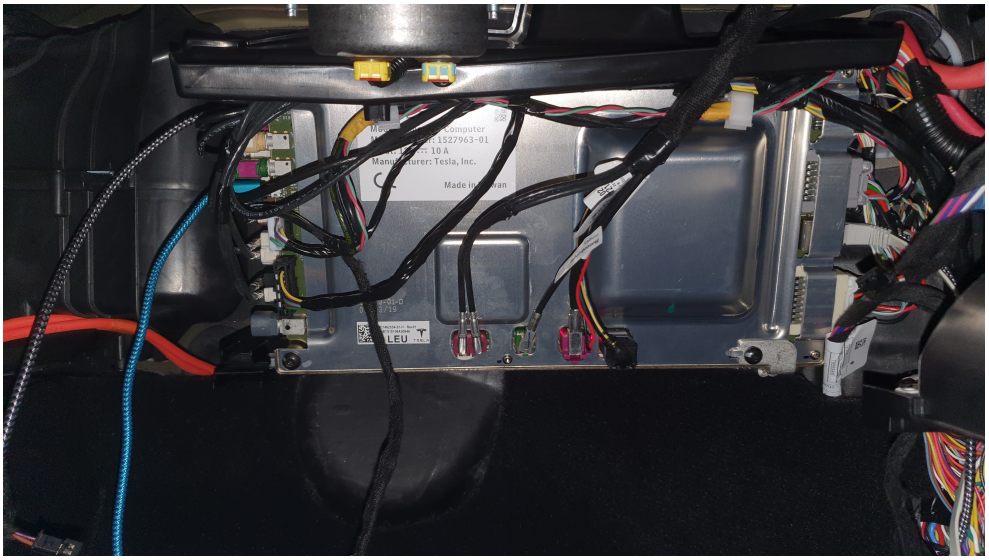

Figure 3.5: MCU unit

ACU unit as it is behind MCU unit. There are many cables connected to the MCU unit.

Chapter 4

Threat modeling

4.1 Assets

We have set our priorities on the meeting with the owner of the car. He defined his assets as following:

1. Car as whole.
2. Personal data stored in the car.

Any unauthorized access to the car can cause potential harm. Unauthorized access can give attacker an oportunity to steal things from the car or even the whole car. It can also lead to damage on the car or even malfunctioning systems, including safety components of the car, what can threaten health of passengers.

Personal data have incalculable value for owner. Therefore attacker has big interest in getting access to these data. Keeping personal data safe is priority for owner of the car.

4.2 Processes

After defining valuable assets I need to identify processes that involve them. I divide these processes into two categories:

1. Human interaction - Usage of the car by people.
2. Processes in the car - Everything that works on the background of the car.

Both categories are important and have effect on defined assets. Losing key fob can have same result as usage of insecure technologies and protocols.

4.3 Attack surface

To identify threats for defined assets I need to map attack surface of the car first. I am mapping attack surface from different points of view:

1. Human aspect
2. Network topology

4.3.1 Human aspect

Users of the car always represent security threat, because they affect car security in an essential way. User has all tools to use the car as he should. Therefore getting these tools from car user is attractive for attacker. User directly affects 4 objects as shown on figure 4.1. Each of mentioned is affected

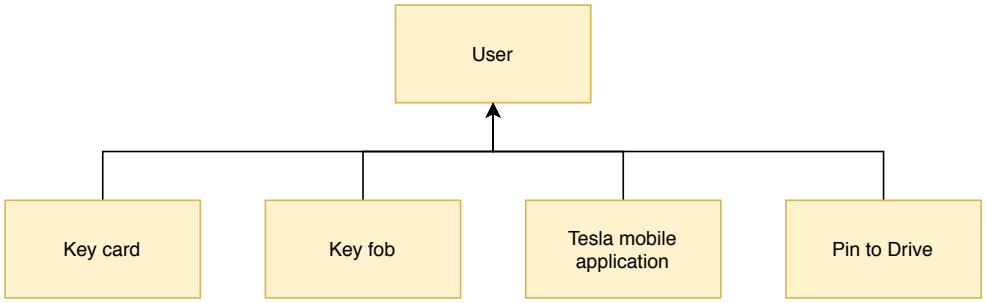

Figure 4.1: User access

in different way.

1. Key card - User can loose his key card from vehicle as well as it can be stolen from him. If attacker gets this key, he will have full control of the car.
2. Key fob - Situation with key fob is similar to key card. It can be lost or stolen.
3. Tesla mobile application - Car can be controlled by Tesla's mobile application. Gaining access to the mobile phone can lead to breaking into the car. Therefore user's phone must be secured as well.
4. Pin to Drive - Pin to Drive is security feature that allows to start the car only after entering a pin. If user chooses trivial pin, this feature has no real security effect.

4.3.2 Network topology

Car uses different networks and its scheme is quite complex at the first glance as shown in the figure 3.1. As this is graph from public source it only gives me clearer image on how network of the car looks.

Car has two types of networks: CAN network and Automotive Ethernet network. CAN network does encrypt communication by design 3.1.4. Connecting to this network means that attacker can see all traffic. He can also send messages to the network if he wants. Car uses 3 main separated CAN networks connected via security gateway. These networks are used for parts that control driving and safety of the car. Automotive Ethernet network is used everywhere else. It is used for connecting cameras to ACU unit and also as a connection between some units.

Connection types There are few ways how to connect to the car. Every type of connection technology has its own usage and function in the car. Car uses LTE, Wi-Fi, Bluetooth connection and radio tuner.

Bluetooth has 2 main functions:

1. It communicates with the key-fob.
2. It is used to connect mobile phones for hands-free calls and sharing multimedia with infotainment system. Mobile phone can be also used as key to unlock the car.

Wi-fi is used to connect to the internet. As there must be available Wi-Fi network to connect to, this connection is meant to be used when the car is parked in owner's garage or when it is in the service. Most of the software updates can be downloaded only via Wi-fi connectivity.

LTE connection is used to provide all multimedia features and map navigation. LTE is used for the connection all the time, except time when the car is connected via Wi-fi.

Radio tuner is used to receive radio waves and provide radio connection of FM and AM waves.

Used devices Identifying devices that could be used to connect to car's network is next step in threat modeling process. Individual devices are shown on figure 4.2. Following table 4.1 describes possible threats how attacker can access CAN network.

Next table 4.2 describes possible threats how attacker can access Ethernet network.

4.3.3 Other threads

There are also other threads that are not directly related with networks. There is a possibility that someone will copy NFC key card to access the vehicle. Also

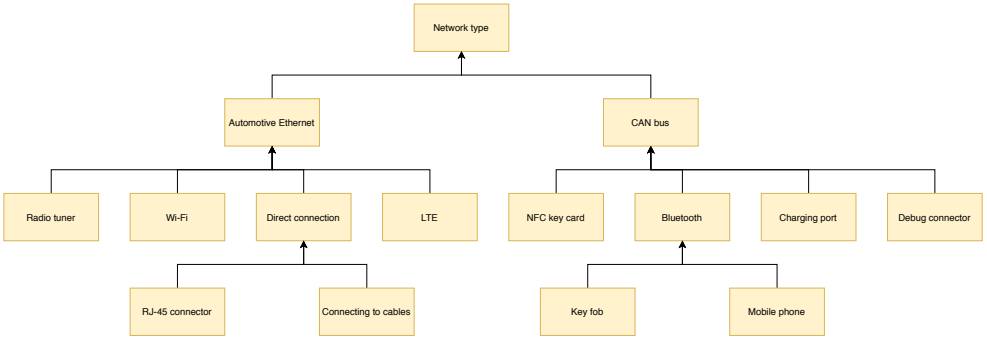

Figure 4.2: Topology of network devices

Access device	Possible threats
Charging connector	There is a communication between the charging station and the car. This communication should not allow the station to send messages directly to one of the main CAN networks.
NFC key card reader	NFC key card reader is essential part of the security infrastructure of the car. If attacker can get physical access to the reader he can send arbitrary value to the security controller. He can also spoof communication between reader and the controller. Convincing security controller leads directly to gaining unauthorized access to the car.
Debug connector	Connecting to CAN network directly through connector means attacker can send whatever message. This can lead to unauthorized access to the car or even unexpected behavior. He can also spoof all the communication on the network.

Table 4.1: Ways to access CAN network

there is a possibility of breaking security challenge when opening the car. Then he will be able to open the car with his own created NFC card. Key fob must be also secured and shouldn't be easily copied. Security function: If cellular signal gets jammed and car has no connection, it's safety functions like SOS call will not work.

4.3.3.0.1 Scope of threads for this work My work focuses on Automotive Ethernet network of the car and possible ways of connecting to it as shown on the figure 4.3. Scope of my work includes analyzing possible threats that relate to Automotive Ethernet network. Devices marked as green are part of the scope. Analysis of CAN network and Bluetooth connectivity is

Access device	Possible threats
Wi-Fi	Wi-Fi connection can be attacked by creating Wi-fi hotspot and listening to all communication between car and internet. Also services available from outside can be attacked.
LTE	Same threat exists for LTE connectivity. Someone can create his own network and then sniff all traffic. He can also connect to available services of the car.
RJ45 connector	Connecting directly to the network using RJ45 connector on the MCU represent threat. It has many potential usages for the attacker from spoofing network to taking control of the car.
Connecting to cables	Connecting directly to the Ethernet network cables represent significant threat. This method potentially enables everything like connecting to RJ45 connector. In addition, attacker can filter communication between units.

Table 4.2: Ways to access Ethernet network

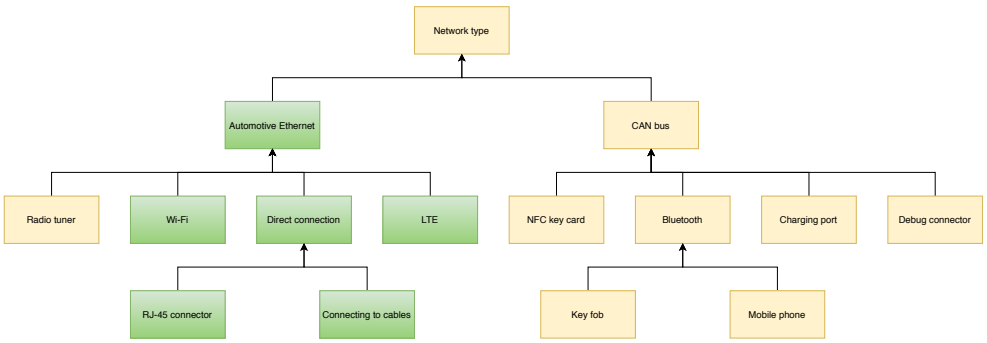

Figure 4.3: Scope of network devices

out of the scope.

Chapter 5

Vulnerability analysis

After identifying threats, that threaten the assets, I need to analyze the vulnerabilities which can lead to exploitation. I begin with the wireless networks (Wi-Fi and LTE network) as they represent bigger risk. After that I analyze vulnerabilities of Automotive Ethernet which is accessible by using RJ45 connector on the MCU unit. Analysis of the direct connection to the cables is the last part. At the end of the chapter summary of found vulnerabilities is described.

5.1 Wi-Fi connection analysis

As described in the intelligence gathering chapter 3.1 Wi-Fi connection is meant to be used when the car is parked somewhere. Analyzing communication security is very important. Creation of Wi-Fi access point is needed in order to analyze this connection. For this purpose, I created Wi-Fi access point by using Linux Mint 19.3 and Lenovo Thinkpad t480s laptop.

First attempt to connect the car to the created Wi-Fi access point was not successful. The car had connected to the access point but immediately after connecting, it checked for internet connection. This is realized by GET request using HTTP protocol to the address `http://connman.vn.tesla.services/online/status.html`. As the access point was not configured to route traffic to the internet, it has disconnected. Modification of the setup was needed. After proper networking configuration of the access point car connected successfully. Access point uses private range 10.42.0.0/24.

With everything set-up I used program Nmap scan to identify connected devices. Configuration of the network with connected devices is shown on table 5.1. Port scan of the available address discovered, that there is one available port 8002 using tcp protocol. It is in state filtered. This port is registered for service teradataordbms in TCP port registry [19], although this port can still be used for any other service. Next step is passive listening to

Device	IP address	MAC address
Wi-Fi access point gateway	10.42.0.1	04:d3:b0:c3:63:56
The car	10.42.0.53	CC:88:26:04:4F:E3

Table 5.1: Available devices to Wi-Fi

identify traffic that goes through this connection. As the car is connected to my access point I got myself into the man in the middle position, so I can sniff whole communication between car and the internet.

Boot process To map communication of the car properly, I connected car to the Wi-Fi network and restarted car's touchscreen using instructions described in the owner's manual [16]. After restart car automatically connected to the Wi-Fi network. Its first step was the same as it checked the internet connection using GET request to an address on Tesla's server. Response to this request is only empty html site. This is really straightforward and functional check. After successful control car connected to Tesla's services using encrypted communication. All communication that goes through this tunnel is encrypted and its content is secured. Then car searched for NTP servers using the DNS queries and after response a device with private IP address `192.168.90.100` sends actual time using the NTP protocol to two servers. Then there is more communication with NTP servers to provide accurate time. Then the car shortly communicated with Google's and Spotify's servers. Both used secured communication. After that the car communicated with Tesla's services using encrypted communication. This communication just went on.

Found local address `192.168.90.100` needs to be analyzed in more details.

User actions Identifying user actions that uses Wi-Fi connection was the next step. By using multimedia system and capturing traffic at the same time I identified following processes:

1. Using browser on the main screen
2. Using maps
3. Using Spotify to play music
4. Downloading software updates

Apart from these processes there is also other ongoing communication which needs further analysis. Each process is analyzed in detail in following paragraphs.

Using browser User can browse internet sites using the internet browser installed in the multimedia system. Connection to web sites is direct. No additional security layer like VPN is used here. Connecting to site using plain http protocol means, that Wi-Fi network can see all the traffic. That exposes user to different MITM attacks like sniffing, ssl stripping, etc.

Using maps Maps can be used for navigation or just to search for places. By sniffing the communication while using the map I found that the car communicates with Google's servers. That means the car is using Google maps. All communication is encrypted using TLS protocol in version 1.3.

Using Spotify Spotify is installed in the multimedia system and is used for streaming music. I identified its traffic by playing different songs while sniffing whole traffic. It uses mix of encrypted and unencrypted traffic. Firstly, there is encrypted communication, that starts new unencrypted communication. This looks like the communication with server is secured and the following stream of songs is realized by plain TCP stream. It doesn't use encryption to secure its traffic, however Spotify encrypts data before sending them to TCP stream, so attacker has still needs to decrypt data in order to access them [20].

Downloading software updates Downloading software updates can not be triggered at any moment. To start this process, there must be some new update waiting for download and multimedia system must request its installation. Whole process starts after user confirms it. I captured the traffic while downloading minor update called Game Update.

5.2 LTE connection analysis

Car uses LTE connection most of the time. It uses LTE connection whenever, there is no Wi-Fi network available as described in intelligence gathering chapter 3.1. This provides car nonstop access to the internet. That is also my first concern if is car available from the internet.

5.2.1 Availability from the internet

Tesla provide mobile application with many functions for better user experience of car. These functions include viewing the vehicle's estimated range, locking or unlocking doors and trunk remotely and so on. Internet connection is required in order to make these functions work.

With connected car to the internet via LTE network I got its public IP address using browser in the multimedia system using public ip finder sites. First attempt was sending ping requests to this address. No response was received from the car. Next attempt was performing port scan on this IP

address. Using Nmap I found open tcp port 1720 as shown on listing 5.1. This is potential vulnerability if attacker is able to use this port to gain access to the car. Investigation of possibility to exploit this port is out of the scope of this thesis.

Listing 5.1: Nmap scan of car's public IP address

```
nmap -Pn 188.207.85.178

Starting Nmap 7.60 ( https://nmap.org ) at 2020-05-27 15:58 CEST
Nmap scan report for static.kpn.net (188.207.85.178)
Host is up (0.042s latency).
Not shown: 999 filtered ports
PORT STATE SERVICE
1720/tcp open h323q931

Nmap done: 1 IP address (1 host up) scanned in 89.27 seconds
```

Whole analysis of the LTE connection was done with my co-worker on this project. Main purpose of this testing is to test security of the car connected to the LTE network not to test the LTE network security as the LTE network is very complex. To be able to describe connection I must provide simplified description of its security architecture first [21].

5.2.2 LTE security architecture

LTE is a cellular network that provides consistent Internet Protocol between an end user's mobile device and IP services on data network. The LTE network has four base parts: User Equipment (UE), base stations, core network and IP network. UE connects to the base station that makes up the E-UTRAN via radio signals, and the base stations transmit and receive IP packets to and from the core network. The core network has large number of entry and exit points including the internet.

Hardware security The Universal Integrated Circuit Card is the next-generation Subscriber Identity Module (SIM) card used in mobile devices. The UICC hosts the Universal Subscriber Identity Module (USIM) application. It performs security critical operations required of LTE cellular networks, such as authentication and other cryptographic functions. One of the most important functions of the UICC is cryptographic key and credential storage. UICCs are provisioned with a long-term, pre-shared cryptographic key. This key is stored within the UICC and also within the core network and never leaves either of those locations. All other keys in LTE's cryptographic structure are derived from this key. The UICC also stores the International Mobile Subscriber Identity (IMSI) and International Mobile Equipment Identifier (IMEI), which are both used to support the use of identities.

UE Authentication The Authentication and Key Agreement (AKA) protocol is the primary LTE authentication mechanism mobile handsets use to authenticate to an LTE network. The AKA protocol cryptographically proves that the UICC and the Mobile Network Operator (MNO) have knowledge of the secret key and provides mutual authentication between the UICC and the LTE network. AKA begins by a User equipment (UE) providing its identifier to the appropriate Mobility Management Entity (MME). After the identifier is provided to the core network, the MME provides the identifier, alongside additional cryptographic parameters and the serving network ID (SN id), to the Home Subscriber Server (HSS)/Authentication Center(AuC). These values are used to generate authentication vector (AUTN) and the expected result (XRES). This authentication vector is then passed back to the MME for storage. The MME provides the AUTN and random parameter (RAND) to the UE, which is then passed to the Universal Subscriber Identity Module (USIM) application. The USIM sends AUTN, RAND, the secret key K, and its Sequence Number (SQN) through the same cryptographic function used by the HSS/AuC. The result is labeled as RES, which is sent back to the MME. If the XRES value is equal to the RES value, authentication is successful and the UE is granted access to the network.

Air Interface Security The UE and the eNodeB communicate using a Radio Frequency (RF) connection commonly referred to as the air interface. Both endpoints modulate IP packets into an RF signal that is communicated over-the-air interface. These devices then demodulate the RF signal into IP packets understandable by both the UE and Evolved Packet Core (EPC). The eNodeB routes these packets through the EPC while the UE uses the IP packets to perform some function. This over-the-air communication is not necessarily private, meaning anything within the wave path can intercept these radio waves. Communication on the air interface can be confidentially protected, but this is left as optional. Air interface confidentiality provides a higher level of assurance, that the messages being sent over the air cannot be deciphered by an external entity.

Backhaul Security The radio access network and associated interfaces make up the E-UTRAN portion of the LTE network. This is the midway between a handset and Mobile Network Operator's core network. Handover is one of the most important functions of a cellular network. There are two types of handover: X2 handover and S1 handover. The transport mechanism between the eNodeB and the EPC is all IP based communications.

As stated in the LTE specifications, security for native IP-based protocols shall be provided at the network layer. This specifications document introduces the notion of Security Domains and using Security Gateways (SEG) or firewalls at the edge of these domains in order to provide security. Security

domains are "networks that are managed by a single administrative authority". Confidentiality is provided by initiating an IPsec tunnel at the eNodeBs for traffic traveling over the (potentially not physically secure) S1 interface and terminating the tunnel at the security gateway placed at the edge of the Security Domain where the EPC is hosted. The use of IPsec on the S1 interface will require endpoints terminating the IPsec tunnel to be provisioned with pre-shared keys or digital certificates.

5.2.3 LTE security testing

To be able to analyze attack surface of the car connected to LTE network we need to connect the car to our created LTE network to get to man in the middle situation, equally to setup when connecting to Wi-Fi network.

To connect car to the LTE network my co-worker prepared fully functional LTE network with configured SIM card. This network is built on open source project called OpenBTS [**?**]. Then we need to bring car's antenna to isolated environment with the created network so the car can not find any other networks to connect to. We also need to insert configured SIM card to the SIM slot on the MCU unit. This connector is located at the bottom of the MCU unit. With this configuration everything should work. SIM slot was empty by default, what suggested that it is not used and other, probably integrated SIM card, is used. Despite that, we prepared the isolated environment with our LTE network, moved car's antenna there and restarted touchscreen. Car tried to connect to our network, because it was the network with the most powerful signal strength. As our SIM card had not been used, car did not connect to our network for obvious reason. After unsuccessful attempt to connect to our network car connected to normal LTE network even without antenna. We tried multiple restarts and even after unplugging whole unit from power SIM card slot was not used.

After unsuccessful first attempt, we tried downgrade attack [21]. My co-worker prepared 2G network in the same isolated environment. We moved car's antenna to this environment and restarted the touchscreen. This time the car did not even try to connect to 2G network. This seems like the car does not support 2G networks.

5.3 MCU's RJ45 connector analysis

Connecting directly to Automotive Ethernet using connector on the MCU unit is a simple process.

Setup Using knowledge from 3.1 I configured IP address to 192.168.90.104, set default GW to 192.168.90.100 and connected to the RJ45 connector on the MCU unit. Network should use network range 192.168.0.0/16 as found from

public source 3.1.3. As this information is acquired from non credible source, I need to confirm this information.

Scanning To identify all available devices arp-scan is used. I include whole output of the command shown on figure 5.2. This result confirms expected results, that configured ip address is working. Output tells me, that switch has MAC address a4:34:d9:01:02:03 and two ip addresses: 192.168.20.2 and 192.168.90.100. No other devices are available. Next step is to get information about open ports available on these addresses. For this purpose I use network scanner called Nmap.

As a result there are two open ports on address 192.168.90.100 5.3 and zero on address 192.168.20.2 5.4. These are standard ports used for SSH connection and web services. These open ports should be definitely tested in next steps.

Listing 5.2: arp-scan of network 192.168.0.0/16

```
arp-scan 192.168.0.0/16

Interface: enx00e04c356537, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9 with 65536 hosts (http://www.nta-monitor.
    ↪ com/tools/arp-scan/)
192.168.20.2 a4:34:d9:01:02:03 (Unknown)
192.168.90.100 a4:34:d9:01:02:03 (Unknown)

2 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 65536 hosts scanned in 263.825 seconds
    ↪ (248.41 hosts/sec). 2 responded

sudo arp-scan 172.16.0.0/12
Interface: enx00e04c356537, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9 with 1048576 hosts (http://www.nta-monitor
    ↪ .com/tools/arp-scan/)

10 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 1048576 hosts scanned in 4198.099 seconds
    ↪ (249.77 hosts/sec). 0 responded
```

To further analyze available devices I captured communication while being connected to this connector. There is a communication going on. All captured packets use ARP protocol. Device with IP address 192.168.90.100 is constantly asking for other devices MAC addresses 5.1. This looks like some kind of check if that device is online. No other communication between units is available which means there might be some kind of filtering used on this connector. Open ports 22 and 8080 on device 192.168.90.100 are my next focus for the tests. On port ssh server is running . Using debug output 5.5 of the ssh

Listing 5.3: Nmap scan of 192.168.90.100

```
nmap -Pn -O 192.168.90.100

Starting Nmap 7.60 ( https://nmap.org ) at 2020-03-12 13:43 CET
Nmap scan report for _gateway (192.168.90.100)
Host is up (0.00084s latency).
Not shown: 998 filtered ports
PORT STATE SERVICE
22/tcp open ssh
8080/tcp open http-proxy
MAC Address: A4:34:D9:01:02:03 (Intel Corporate)
Warning: OSScan results may be unreliable because we could not
    ↪ find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.10 - 4.8, Linux 3.2 - 4.8
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at
    ↪ https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.70 seconds
```

Listing 5.4: Nmap scan of 192.168.20.2

```
nmap -Pn -O 192.168.20.2

Starting Nmap 7.60 ( https://nmap.org ) at 2020-03-12 14:34 CET
Nmap scan report for _gateway (192.168.20.2)
Host is up (0.0012s latency).
All 1000 scanned ports on _gateway (192.168.20.2) are filtered
MAC Address: A4:34:D9:01:02:03 (Intel Corporate)
Too many fingerprints match this host to give specific OS
    ↪ details
Network Distance: 1 hop
```

```
60 339.880026449 IntelCor_01:02:03    Broadcast           ARP         60 Who has 192.168.90.105? Tell 192.168.90.100
61 340.075175899 IntelCor_01:02:03    Broadcast           ARP         60 Who has 192.168.90.60? Tell 192.168.90.100
62 340.227606813 IntelCor_01:02:03    Broadcast           ARP         60 Who has 192.168.90.30? Tell 192.168.90.100
63 342.670296035 IntelCor_01:02:03    Broadcast           ARP         60 Who has 192.168.90.102? Tell 192.168.90.100
64 342.670323818 LcfcHefe_a4:22:55    IntelCor_01:02:03   ARP         42 192.168.90.102 is at 98:fa:9b:a4:22:55
65 424.217473974 IntelCor_01:02:03    00:55:7b:b5:7d:f7   ARP         60 192.168.90.100 is at a4:34:d9:01:02:03
66 435.129355236 IntelCor_01:02:03    Broadcast           ARP         60 Who has 192.168.20.1? Tell 192.168.20.2
67 440.241217473 IntelCor_01:02:03    00:55:7b:b5:7d:f7   ARP         60 192.168.20.2 is at a4:34:d9:01:02:03
68 457.206839328 IntelCor_01:02:03    TeslaMot_01:02:03   ARP         60 Who has 192.168.90.102? Tell 192.168.90.100
69 470.312640267 IntelCor_01:02:03    00:55:7b:b5:7d:f7   ARP         60 192.168.20.2 is at a4:34:d9:01:02:03
70 498.139961932 IntelCor_01:02:03    00:55:7b:b5:7d:f7   ARP         60 192.168.20.2 is at a4:34:d9:01:02:03
71 530.786168904 IntelCor_01:02:03    00:55:7b:b5:7d:f7   ARP         60 192.168.20.2 is at a4:34:d9:01:02:03
```

Figure 5.1: Sample of captured ARP packets

Listing 5.5: SSH configuration analysis

```
ssh -vN tesla@192.168.90.100

...
debug1: Remote protocol version 2.0, remote software version
    ↪ OpenSSH_7.9
...
kex_input_ext_info: server-sig-algs=<ssh-ed25519,ssh-rsa,rsa-
    ↪ sha2-256,rsa-sha2-512,ssh-dss,ecdsa-sha2-nistp256,ecdsa-
    ↪ sha2-nistp384,ecdsa-sha2-nistp521>
...
```

command I discovered, that it uses standard OpenSSH library in version 7.9 and displayed cryptographic methods. Public vulnerability database reports 4 vulnerabilities for this version [22]. All of these vulnerabilities can harm only client in case of bad server configuration.

5.3.0.0.1 MCU Switch analysis After initial connector testing I started to analyze configuration of the switch in cooperation with my project coworker. At first, I need to identify devices connected to the switch. Using connection schema of the switch 5.2 there are 7 devices connected named P0, P1, P2, P3, P4, P5 and P6. Next step is to match these with real devices. After analyzing the connection I identified all devices:

1. P0 - Intel Atom
2. P1 - LTE modem
3. P2 - U22 100BASE-T1
4. P3 - J3 100BASE-TX
5. P4 - J15 RJ45
6. P5 - Gateway

7. P6 - U23 100BASE-T1

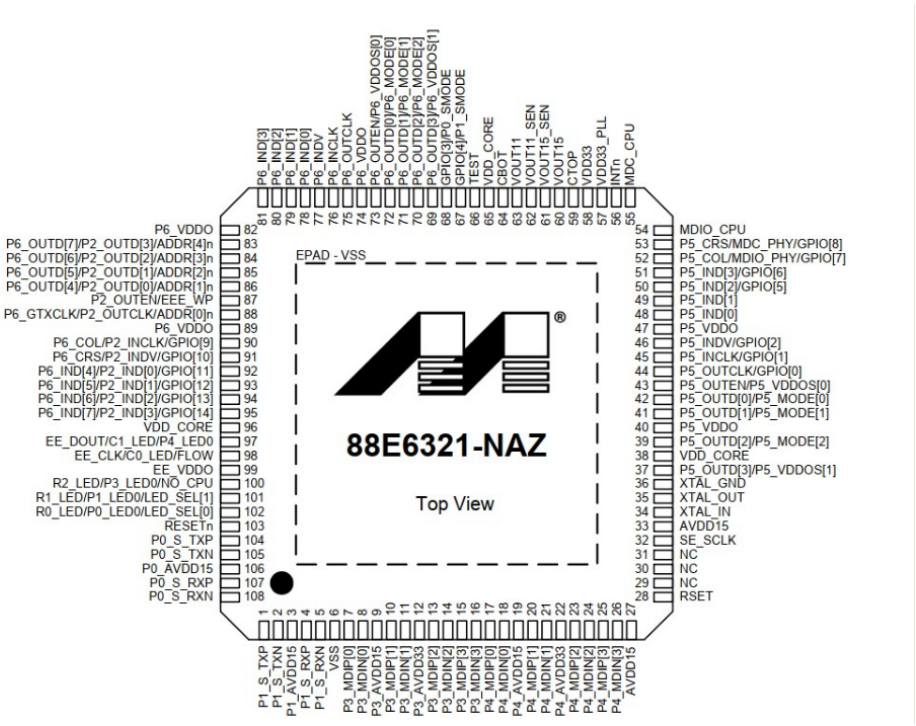

Figure 5.2: Switch internal connection[23]

Configuration of the switch Next step is configuration analysis of the switch. This switch is configured every time it is turned on. By sniffing communication between switch and Intel Atom CPU we got long log. This sniffing was done by my co-worker, who is working on analyzing hardware of the switch.

Analyzing the log. Format of the log looks like this Wa16r06_0019. Where the first letter W means "write operation to register" and R means "read operation from register". a16r06 means SMI address 0x16 at register 06. Last four digits are written or read data. I was able to identify all ports configuration using the register configuration data sheet. 5.3 All seven ports have the same configuration. Important part is, that all ports use VLANs defined in VLANTable, 802.1Q VLANs defined in the VTU (if 802.1Q is enabled) and Trunk Masking are enforced for ALL frames. VLAN mapping is defined in register 0x06 for each port. For example, port P4 has a value of the register 0x006F. Converted to binary 1101111. Fifth position, where the value is zero, belongs to this port. This means this port can send frames to all other ports. Table 5.2 represents VLAN port configuration. This tells me that all other

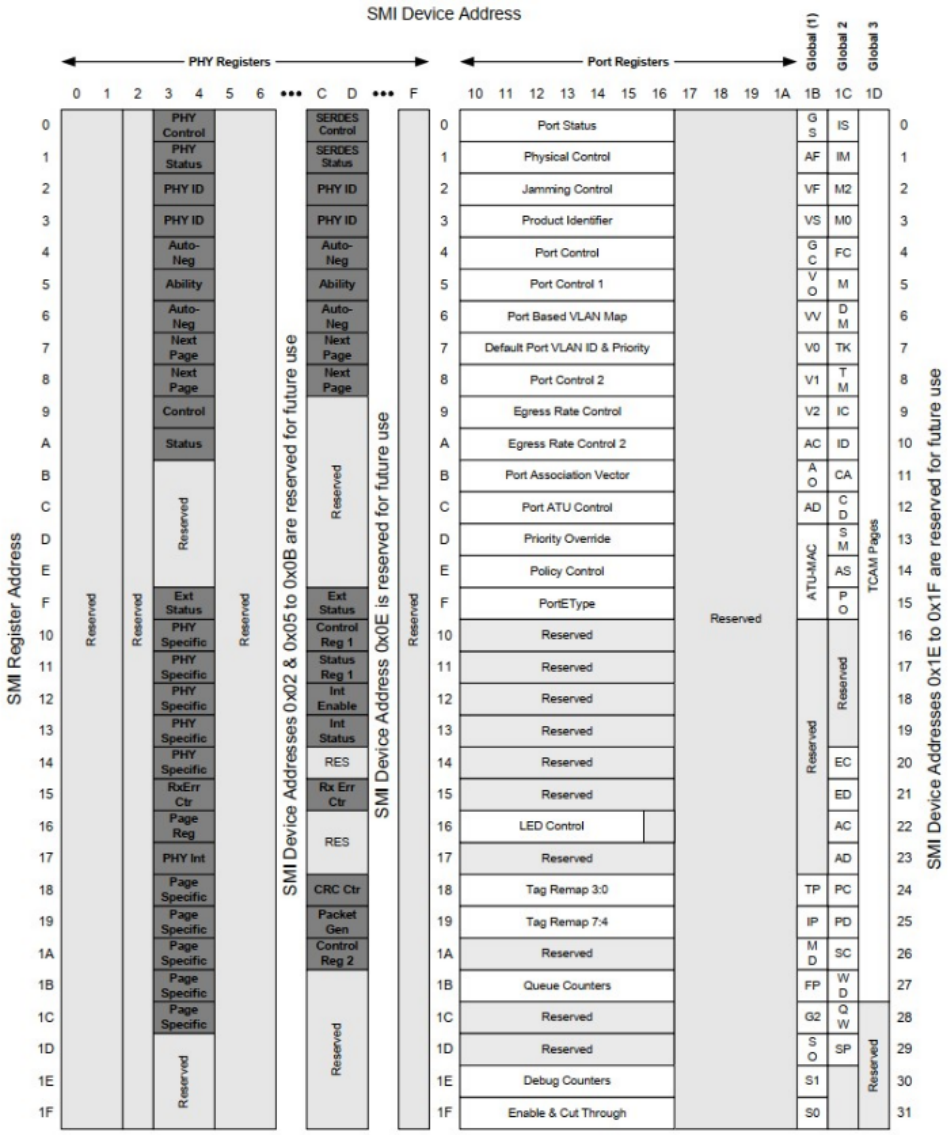

Figure 5.3: Switch registers[23]

devices should receive packets sent from RJ45 connector, that was used in the previous tests. This means there should be other filtering of the traffic. All this indicates that switch uses ATU table to filter traffic.

Remote Management Remote Management [24] allows to configure switch using Ethernet frames. It can access registers of the switch. There is a Remote Management Unit (RMU) for this reason. It can be seen on the switch configuration diagram 5.4 as part of the switch configuration.

	P6	P5	P4	P3	P2	P1	P0
P6	-	X	1	1	X	X	1
P5	X	-	1	1	X	X	1
P4	1	1	-	1	1	1	1
P3	1	1	1	-	1	1	1
P2	X	X	1	1	-	X	1
P1	X	X	1	1	X	-	1
P0	1	1	1	1	1	1	-

Table 5.2: VLAN port configuration

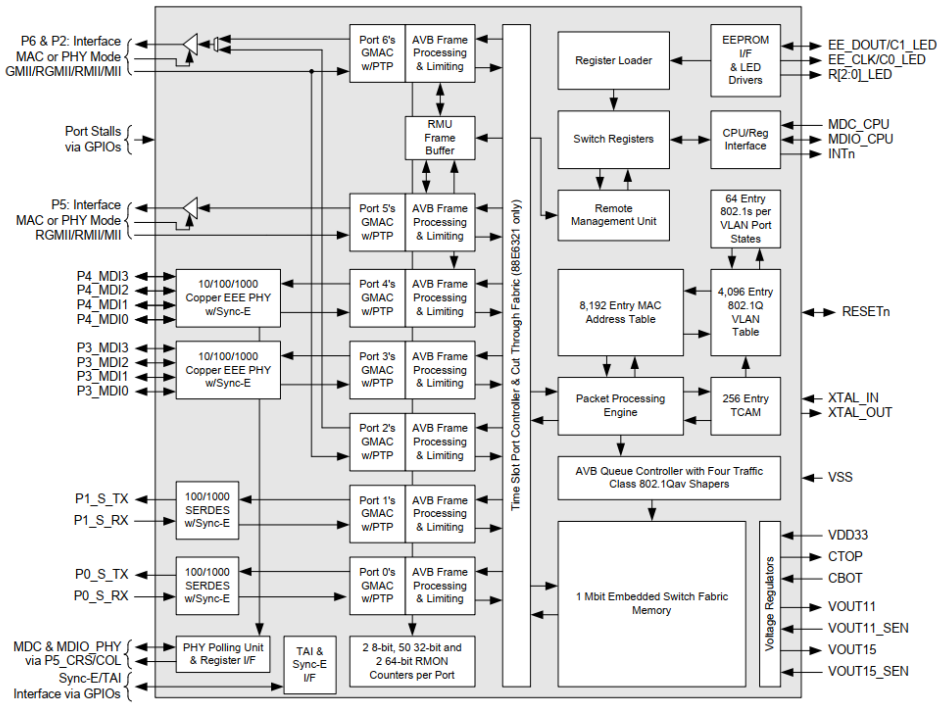

Figure 5.4: Switch internal configuration[23]

5.4 BroadR-Reach analysis

BroadR-Reach is widely used in the internal network of the car. To connect classic computer to BroadR-Reach be able to communicate on TCP/IP stack dedicated hardware is needed. For this purpose FC602 USB-Stick was used [25]. Without dedicated hardware connection to BroadR-Reach is not possible what makes it unreachable for most enthusiast.

5.4.1 Setup

Proper setup to connect to BroadR-Reach and maintain everything in the car working is needed. BroadR-Reach uses one twisted pair for both directions of communication. Simple cutting cable and connecting it to the USB-Stick would break connectivity between units. To connect to the cable bridge is needed to ensure origin connection is not interrupted.

Two bridges were created. First between the ACU unit and the MCU unit as these are the most important units. Cable between ACU and MCU was interrupted and each end was connected to FC602 USB-Stick [25]. Both USB-Sticks were connected to Raspberry Pi 3 computer and configured as network bridge. To be able to communicate on TCP/IP stack, IP address was configured to 192.168.90.104. Second bridge was configured between MCU unit and radio tuner. Here was configured IP address 192.168.90.29. To route traffic correctly first bridge routes traffic for network range 192.168.90.64/26 and second for network range 192.168.90.24/29. This was the most straightforward way to get everything working. This configuration was possible only because there is a big gap of used IP addresses between 192.168.90.30 and 192.168.90.100. Static routes would be needed in other case. This configuration provided stable communication between units and all systems in the car worked as intended without breaking. Final configuration is shown on figure 5.5.

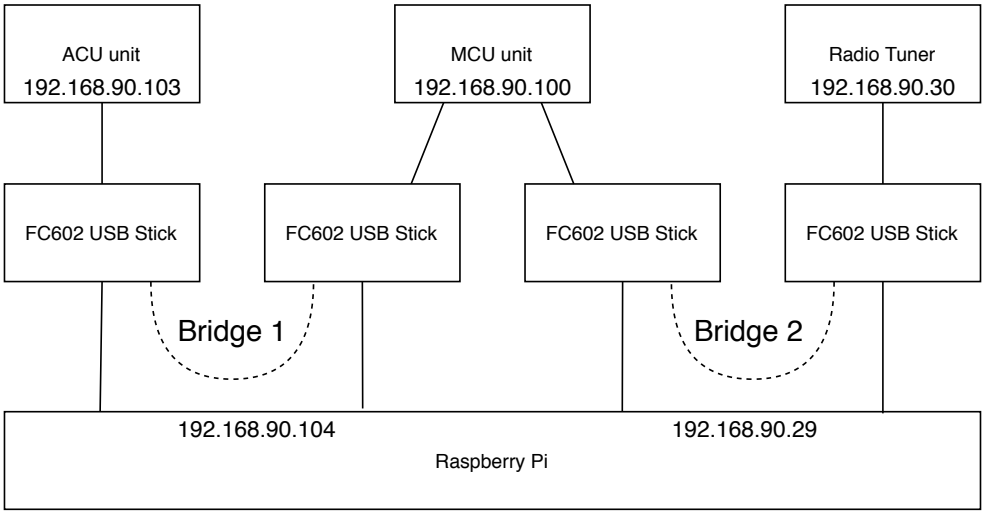

Figure 5.5: BroadR-Reach connection setup

5.4.2 Scanning

Identifying available devices is the first step. I used Nmap for this purpose as in the previous cases. There are four devices available with IP addresses:

Device	IP address	MAC address
MCU unit	192.168.90.100	A4:34:D9:01:02:03
ACU unit	192.168.90.103	00:02:5A:C5:4F:00
ACU unit	192.168.90.105	00:43:58:85:4C:02
Radio tuner	192.168.90.30	2C:6B:7D:C4:41:3F

Table 5.3: Available devices on BroadR-Reach network

192.168.90.100, 192.168.90.103, 192.168.90.105 and 192.168.90.30. Available devices on network are shown in table 5.3.

Next step is to identify open ports and available services on these devices. Again Nmap is good tool for this. Starting with MCU gateway I found many available ports that are open or filtered as shown on figure 5.6. Same scan was repeated for rest available devices. All other devices have available same ports: 22, 8901, 25974, 28496. Only one figure is shown as other was exactly the same 5.7.

Difference is significant in comparison with scans of the same devices from debug connector 5.3 . There are many more open ports. This confirms my theory, that the debug connector is using some kind of filtering communication with other devices on the network. On the other hand devices fully trust BroadR-Reach network and exposes many services. Therefore more analysis is needed to investigate present threads.

5.4.3 Passive listening

Passive listening provides information about normal communication on the network. This captures communication between units in normal operation without any modifications. With turned on car I started capturing traffic for further analysis. I captured traffic on both configured bridges.

Bridge 1 Captured traffic provided essential information about communication between units. First finding is that no secure protocols are used for communication. Units use only plain TCP, HTTP and UDP to communicate with each other. This brings same vulnerability that is characteristic to CAN bus networks. Anyone on the network sees whole traffic of everyone else. Not only that any device can pretend to be other device and sends any packet to network as it. These vulnerabilities are therefore same as on the CAN bus network and change to the Ethernet based network has not solved it.

Form of communication is also interesting. Basically there are two types of communication on local network:

1. Direct TCP communication between units.

2. Unit sending packets as UDP broadcast.

Listing 5.6: Nmap scan of 192.168.90.100

```
nmap -p- 192.168.90.100
Starting Nmap 7.80 ( https://nmap.org ) at 2020-04-28 17:05 UTC
Nmap scan report for cid (192.168.90.100)
Host is up (0.038s latency).
Not shown: 65505 closed ports
PORT STATE SERVICE
22/tcp filtered ssh
3490/tcp open colubris
4030/tcp open jdmn-port
4032/tcp open vrts-auth-port
4035/tcp open wap-push-http
4037/tcp open ravehd
4050/tcp filtered cisco-wafs
4060/tcp open dsmeter_iatc
4070/tcp open tripe
4080/tcp open lorica-in
4082/tcp filtered lorica-out
4090/tcp filtered omasgport
4094/tcp filtered sysrqd
4096/tcp filtered bre
4110/tcp open g2tag
4160/tcp filtered jini-discovery
4170/tcp open d-cinema-csp
4210/tcp filtered vrml-multi-use
4220/tcp open vrml-multi-use
4280/tcp open vrml-multi-use
4400/tcp open ds-srv
4500/tcp open sae-urn
4504/tcp filtered unknown
4506/tcp open unknown
4508/tcp filtered unknown
7654/tcp open unknown
8002/tcp filtered teradataordbms
8080/tcp filtered http-proxy
20564/tcp open unknown
25956/tcp open unknown
MAC Address: A4:34:D9:01:02:03 (Intel Corporate)
```

Listing 5.7: Nmap scan of 192.168.90.103

```
Nmap scan report for 192.168.90.103
Host is up (0.0035s latency).
Not shown: 65531 filtered ports
PORT STATE SERVICE
22/tcp open ssh
8901/tcp open jmb-cds2
25974/tcp closed unknown
28496/tcp open unknown
```

Listing 5.8: Example of message send to UDP broadcast

```
$PTSLA_FASTGPS
    ↪ ,1,50.1074481,14.4429749,359.922444,856.883000,0.000000,
    ↪ 2020,4,22,13,24,49,600000000,359.922444*7D
```

None of the units are limited only to one type of communication, all uses both types. Most of the traffic is UDP broadcast. That means that every unit is sending data and all other units receive this data. If any other unit needs this data it already have. This is similar design as is used in the CAN bus networks. However only small amount data is send in each packet. Each packet contains message in the payload. After analyzing these packets I found that the message in the payload uses following format: identifier always starting with $ and then some kind of message. One of that messages is shown on listing 5.8. Looking on the message in depth, after identifier follows GPS coordinates. This creates another vulnerability that attacker can abuse this information in many ways. If he has remote access to the network he can trace actual position of car. If he is able to monitor movement of the car for longer time he knows when is best time to steal the car, best time to burgle owner's house and so on. I assume that direct communication is used for larger payloads that do not fit into one packet or when specific information is needed.

Besides that all units communicate with external servers. Communication with external servers is secured which is only good.

Bridge 2 Traffic on second also uses only plain protocols as expected after analyzing traffic from bridge 1. Communication between 192.168.90.30 and 192.168.90.100 looks interesting. Communication is not encrypted and messages are very similar. There are more simultaneous conversations between these devices 5.9. Biggest difference is that 192.168.90.30 sends longer data with length from 17 to 32 bytes and 192.168.90.100 always sends 16 bytes of

data. I suppose, this is some kind of setting status and then informing about result as response. This communication does not use any of opened ports scanned previously.

5.4.3.1 24 hour sniffing

Capturing 24 hours of the car's communication was done to identify communication when car is turned off. This sniffing should also capture turn off sequence.

There is encrypted communication from the two devices with addresses 192.168.90.105 and 192.168.90.103 to the internet. After 11 minutes from the start, turn off sequence begins. All encrypted communication is closed and communication ends for many hours. Then only two packets are send to local IP address 192.168.90.60 from 192.168.90.100. These are only SYN packets without any data.

5.4.3.2 Game update

Car wanted to download game update. We prepared setup, started capturing communication and then triggered installation of this update. We started sniffing a few seconds before starting the update process and stopped about a minute after it had finished. Communication captured on Wi-Fi is encrypted at first. Then car started downloading update from server using plain TCP connection. Downloaded data does not match any standard format. That means that it is some binary file or that it was encrypted before sending. This is out of scope of this thesis but can be topic of future research.

Listing 5.9: Sample from communication between 192.168.90.100 and 192.168.90.30

```
# Packet 8
peer0_0: !!binary |
AHgH0QAAAAgTQxCVAQAAAA==
# Packet 9
peer1_0: !!binary |
AHgH0QAAAAwTQxCVAQCAAAAAAAI=
# Packet 17
peer0_1: !!binary |
AHgH0QAAAAgTQxCfAQAAAA==
# Packet 18
peer1_1: !!binary |
AHgH0QAAAAwTQxCfAQCAAAAAAAI=
# Packet 23
peer0_2: !!binary |
AHgH0QAAAAgTQxCnAQAAAA==
# Packet 24
peer1_2: !!binary |
AHgH0QAAAAwTQxCnAQCAAAAAAAI=
# Packet 29
peer0_3: !!binary |
AHgH0QAAAAgTQxCvAQAAAA==
# Packet 30
peer1_3: !!binary |
AHgH0QAAAAwTQxCvAQCAAAAAAAI=
# Packet 35
peer0_4: !!binary |
AHgH0QAAAAgTQxC3AQAAAA==
# Packet 36
peer1_4: !!binary |
AHgH0QAAAAwTQxC3AQCAAAAAAAI=
```

Chapter 6

Analysis outcome

Analysis of internal network of the Tesla Model 3 is successful. Analysis describes network configuration, defines threat model and identifies various vulnerabilities.

6.1 Network configuration

Tesla Model 3 uses two types of network: CAN bus and Ethernet network. Ethernet network is analyzed in detail as main goal of the thesis. It uses BroadR-Reach technology to connect units to each other. BroadR-Reach technology is used to connect the MCU unit, ACU unit, radio tuner. As BroadR-Reach uses only one twisted pair to successfully connect to this network, creation of bridge configuration is needed. Connecting to this network also requires dedicated hardware what makes connecting to this network problematic for enthusiasts. Accessibility of this network is also quite awkward as disassembling of dash-board is needed. Another possibility is connecting using debug port on the MCU unit or debug connector located on the bottom of the dashboard on the driver's side.

The car also uses wireless networks using Ethernet protocol: LTE and Wi-Fi. Both are meant to connect the car to the internet and they complement each other.

6.2 Findings

Analysis identifies various vulnerabilities connected with networks. It also identified processes that are secured on high level. Findings are following:

- Car connected via LTE network is visible from the internet. It has open port 1720 what can lead to connection to the car from the internet.

- LTE modem uses integrated SIM what increases complexity of attack with creating simulated LTE network and forces car to connect to it.
- Car does not connect to 2G networks. This is very important because there exists successful attack on 2G networks.
- Connecting the car to Wi-Fi access point exposes user to some risk when using web browser. If user connects to web site using HTTP protocol all traffic is readable by access point. This also exposes user if he connects to web pages using HTTPS without properly configured HSTS header. On the other side all other communication is well encrypted and protected from potentially malicious access point.
- Debug connector located on the bottom side of the dashboard on the rider's side and debug connector on the MCU unit are both properly filtered and no traffic is exposed to attacker. He also can not interact with any device.
- Traffic on BroadR-Reach network is without any protection. Communication on this network is exposed similarly as traffic on older CAN bus network. Despite better technology no encryption is used on local network. To exploit this vulnerability dissembling of the dashboard and dedicated hardware is needed but its potential is still huge.

Chapter 7

Attacks demonstration

This chapter demonstrates possible usage of found vulnerabilities during analysis. With consideration of outcomes of the analysis I prepared two attacks that use different attack vectors and require different access to the vehicle:

1. Man in the middle attack as Wi-Fi access point
2. Sending false GPS data

7.1 Man in the middle attack as Wi-Fi access point

This attack is based on fact that after connecting car to Wi-Fi access point whole traffic from the car flows through it. Attacker can downgrade user's connection to web sites from HTTPS to HTTP if these sites are not configured properly. This exposes user's personal data. Process of this attack is shown on figure 7.1. Purpose of this attack is to show that Wi-Fi access point is able to manipulate traffic of the car and connecting to malicious one can lead to stealing user's personal data.

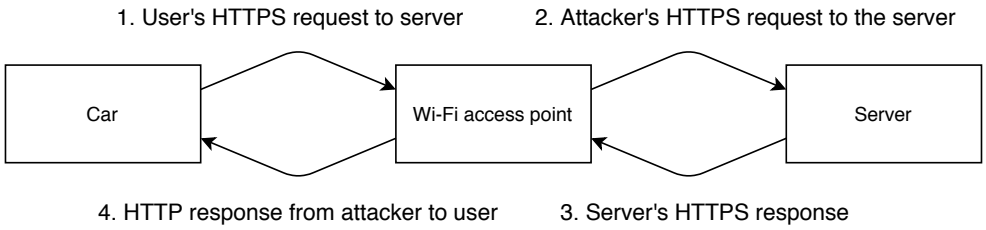

Figure 7.1: SSL striping attack

Setup

Preparing this attack is straightforward. Following steps are required:

1. Create Wi-Fi access point.
2. Redirect HTTP requests to intercept them.
3. Use sslstrip program to downgrade user's connection from HTTPS to HTTP.
4. Monitor user's traffic.

Execution

Execution of this attack is straightforward in the same way as its setup. Only my laptop with internet connection was needed to execute this attack. After initial setup I connected car to created Wi-Fi access point and started browsing web sites. Attack was successful and when user access web pages that do not use HSTS headers web page is loaded using HTTP protocol. Captured traffic shows whole communication in plain text as expected. This attack can be improved as attacker can modify web pages and insert into them whatever he wants.

It turned out Tesla's browser do not allow to view sites with invalid certificate and therefore web sites using HSTS headers were not loaded. That is better as it protects user from stealing his personal data.

7.2 Sending false GPS data

As I discovered during BroadR-Reach analysis 5.4.3 device sends GPS data to broadcast address. As it should do it for some reason some service should use this data. My attack is based on this this prerequisite. Whole idea is to send modified packets with changed location to broadcast. However only coordinates are modified in whole packet, packet still looks like it was sent by the same device. Connection to BroadR-Reach is obviously needed to perform this attack.

Preparation

Firstly I needed to capture packet with GPS message to be able to copy it. To create exactly same packet I used program that is called Scapy [26]. Using this program I was able to create exact copy of the packet with modifying only the coordinates. I prepared script with hard coded values to send packet in loop.

Execution

I turned on the maps on the multimedia center to see my actual position. After being successfully connected to the BroadR-Reach, I started sending packets with modified coordinates. Packets were captured in the traffic but my position on the map shown on the touchscreen has not changed at all. This can be caused by various reasons. For example the touchscreen do not use this coordinates to set position on the map. Despite my unsuccessful attempt I still think that this attack vector is worth more analysis, however that is out of the scope of this thesis but it might be a good topic for future exploration.

Conclusion

Main goal of this thesis was to perform security analysis of the internal network of the Tesla Model 3 with main focus on Ethernet based network.

Analysis was performed based on guideline created and described in the second chapter. I created this guideline by modifying PTES guideline to ensure that analysis is well structured and all important parts are covered. Chapter 3 gathers available information about security of the networks used in the cars. Chapter 4 is devoted to the creation of threat model. Using information from previous chapters I created threat model identifying possible vulnerabilities. I define scope as for further analyzing the vulnerabilities what is done in next chapter. Step by step is every vulnerability in the scope analyzed. LTE and Wi-Fi networks are tested as they provide wireless access to the internal network. Ethernet network is tested via using available connectors and creating direct connection to BroadR-Reach network. Results of the analysis as described in the chapter 6 are:

- Car connected via LTE network is visible from the internet. It has open port 1720 what can lead to connection to the car from the internet.
- LTE modem uses integrated SIM what increases complexity of attack with creating simulated LTE network and force car to connects to it. Car also does not connect to 2G networks.
- Connecting the car to Wi-Fi access point exposes user to some risk when using web browser. If user connects to web site using HTTP protocol all traffic is readable by access point. This also exposes user if he connects to web pages using HTTPS without properly configured HSTS header. On the other side all other communication is well encrypted and protected from potentially malicious access point.
- Debug connector located on the bottom side of the dashboard on the rider's side and debug connector on the MCU unit are both properly filtered.

- Traffic on BroadR-Reach network in without any protection.

In the last chapter attacks demonstrate how vulnerabilities can be exploited. Mitm attack using Wi-Fi access point was successful. Demonstrative attack of sending false GPS data to the local network was not successful, but can be topic of future exploration. Using Ethernet network does not mean that Tesla Model 3 is perfectly secured. Outcomes of this thesis shows that the goal of this thesis was fulfilled.

During working on this thesis I got familiar with specifics of automotive industry and its demands on information technology. I got an opportunity to research a real car, work with its hardware and apply learned theory by performing various attacks. Result of my work is not only this thesis but also my new experience with aquired knowledge apllyied realtime in a real enviroment.

Questions

Bibliography

[1] Schmidt, B. Tesla sells more electric cars than next two biggest EV makers combined [online]. 2020, [cit. 25.5.2020]. Available from: https://thedriven.io/2020/05/04/tesla-sells-more-electric-cars-than-next-two-ev-makers-combined/

[2] Ixia. *Automotive Ethernet: An Overview [online]*. [cit. 10.5.2020]. Available from: https://support.ixiacom.com/sites/default/files/resources/whitepaper/ixia-automotive-ethernet-primer-whitepaper_1.pdf

[3] Dostál, J. Introduction to Ethical Hacking Penetration Testing [online]. 2020, [cit. 29.2.2020]. Available from: https://courses.fit.cvut.cz/BI-EHA/media/lectures/01_Introduction.pdf

[4] OWASP. *Brute Force Attack [online]*. [cit. 29.4.2020]. Available from: https://owasp.org/www-community/attacks/Brute_force_attack

[5] OWASP. *Buffer Overflow [online]*. [cit. 29.4.2020]. Available from: https://owasp.org/www-community/vulnerabilities/Buffer_Overflow

[6] Coursen, S. Safety vs. Security: Understanding the Difference May Soon Save Lives[online]. 2020, [cit. 1.5.2020]. Available from: https://owasp.org/www-community/vulnerabilities/Buffer_Overflow

[7] Miessler, D. Secrecy (Obscurity) is a Valid Security Layer [online]. 2019, [cit. 27.5.2020]. Available from: https://danielmiessler.com/study/security-by-obscurity/

[8] Davis, R.;Burns, A.; Bril, R.; Lukkien, J. Controller Area Network (CAN) schedulability analysis: Refuted, revisited and revised. *Real-Time Systems*, volume 35, no. 3, 2007: pp. 239–272.

[9] Lyon, G. Nmap Network Scanning[online]. 2011, [cit. 1.5.2020]. Available from: https://nmap.org/book/toc.html

[10] Hills, R. The ARP Scanner[online]. 2020, [cit. 1.5.2020]. Available from: https://github.com/royhills/arp-scan#documentation

[11] Tesla. Product Security[online]. 2020, [cit. 1.5.2020]. Available from: https://www.tesla.com/cs_CZ/about/security

[12] PTES. *High Level Organization of the Standard [online]*. [cit. 29.2.2020]. Available from: http://www.pentest-standard.org/index.php/Main_Page

[13] Garcia, D.; Oswald, D.; Kasper, T. Lock It and Still Lose It – On the (In)Security of Automotive Remote Keyless Entry Systems [online]. 2016, [cit. 10.4.2020]. Available from: https://www.usenix.org/system/files/conference/usenixsecurity16/sec16_paper_garcia.pdf

[14] Greenberg, A. Hackers Reveal Nasty New Car Attacks[online]. 2013, [cit. 10.4.2020]. Available from: https://www.forbes.com/sites/andygreenberg/2013/07/24/hackers-reveal-nasty-new-car-attacks-with-me-behind-the-wheel-video/#50904806228c

[15] Greenberg, A. https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/[online]. 2015, [cit. 10.4.2020]. Available from: https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

[16] Tesla. *Model 3 Owner's manual [online]*. [cit. 10.5.2020]. Available from: https://www.tesla.com/sites/default/files/model_3_owners_manual_north_america_en.pdf

[17] Wardell, J. Diagnostic Port and Data Access [online]. 2020, [cit. 5.3.2020]. Available from: https://teslaownersonline.com/threads/diagnostic-port-and-data-access.7502/

[18] nlc. Successful connection on the Model S internal Ethernet network [online]. 2014, [cit. 25.5.2020]. Available from: https://teslamotorsclub.com/tmc/threads/successful-connection-on-the-model-s-internal-ethernet-network.28185/

[19] IANA. *Service Name and Transport Protocol Port Number Registry [online]*. [cit. 22.5.2020]. Available from: https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.txt

[20] Spotify. *Scalable User Privacy [online]*. [cit. 23.5.2020]. Available from: https://labs.spotify.com/2018/09/18/scalable-user-privacy/

[21] Jeffrey Cichonski, M. B., Joshua M. Franklin. Guide to LTE Security? [online]. 2020, [cit. 24.5.2020]. Available from: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-187.pdf

[22] CVE. *Openssh ≫ 7.9 : Security Vulnerabilities [online]*. [cit. 11.5.2020]. Available from: https://www.cvedetails.com/vulnerability-list/vendor_id-97/product_id-585/version_id-274092/Openbsd-Openssh-7.9.html

[23] VesaMount. Marvell switch chip 88E6321 / 88E6320 driver summary-hardware [online]. 2020, [cit. 10.4.2020]. Available from: https://blog.csdn.net/vesamount/article/details/86591448

[24] Marvell. *Link Street® 88E6096/88E6097/88E6097F Datasheet [online]*. [cit. 12.4.2020]. Available from: http://read.pudn.com/downloads218/sourcecode/embed/1024649/88E6095.pdf

[25] FibreCode GmbH. *The USB-Stick for Automotive Ethernet [online]*. [cit. 23.5.2020]. Available from: https://fibrecode.com/download/fc602-usb-oabr-stick/PF_FC602_USB_OABR_Stick_V_01_01_00.pdf

[26] Philippe Biondi and the Scapy community. *Link Street® 88E6096/88E6097/88E6097F Datasheet [online]*. [cit. 28.5.2020]. Available from: https://scapy.readthedocs.io/en/latest/

Appendix A

Acronyms

IoT Internet of things

LTE Long-Term Evolution

BT Bluetooth

OSINT Open source intelligence

CAN Controller Area Network

DHCP Dynamic Host Configuration Protocol

MAC address Media access control address

VIN Vehicle identification number

AWD All-wheel drive

IEEE Institute of Electrical and Electronics Engineers

SMI Serial Management Interface

SIM Subscriber Identity Module

ARP Address Resolution Protocol

IP address Internet Protocol address

GW Gateway

IP Internet Protocol

E-UTRA Evolved Universal Terrestrial Radio Access

Appendix B

Contents of enclosed CD

```
readme.txt ..... the file with CD contents description.
src ..... the directory of source codes
    thesis ..... the directory of LaTeX source codes of the thesis
pcaps ..... the directory of captured pcaps
text ..... the thesis text directory
    thesis.pdf ..... the thesis text in PDF format
```