On Threat Analysis and Risk Estimation of Automotive Ransomware*

Full Paper

Nils Weiss
University of Applied Sciences Regensburg
Regensburg, Germany
Nils2.Weiss@othr.de

Markus Schroetter
University of Applied Sciences Regensburg
Regensburg, Germany
Markus1.Schroetter@st.othr.de

Rudolf Hackenberg
University of Applied Sciences Regensburg
Regensburg, Germany
Rudolf.Hackenberg@othr.de

ABSTRACT

This paper combines a theoretical model for the risk estimation of a ransomware attack on vehicles with our experiences during an implementation of a real world ransomware as proof of concept. Our gained knowledge on ransomware attacks targeting a real car is transferred into a general model for risk estimation. It provides a generic guideline for the risk estimation and allows for identifying possible weaknesses in a vehicle's design concerning the threat of automotive ransomware. Through our abstracted approach, this model is applicable on every modern car. An example to prove this model is provided as well.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability;

KEYWORDS

Ransomware, Automotive Software, Security

ACM Reference format:
Nils Weiss, Markus Schroetter, and Rudolf Hackenberg. 2019. On Threat Analysis and Risk Estimation of Automotive Ransomware. In *Proceedings of ACM Computer Science in Cars Symposium, Munich, Germany, September 2019 (ACM-CSCS'19),* 9 pages.
https://doi.org/10.475/123_4

1 INTRODUCTION

As almost everything in our modern society, vehicles are becoming more digital and connected. They are equipped with increasing numbers of small computers (Electronic Control Units (ECUs)) to improve safety and comfort. This development is an opportunity to improve everybody's life, but can lead to critical threats in terms of security. Security researchers already showed that automotive attacks are possible and may have fatal consequences if they are used for malicious purposes[12].

Looking at the recent trends in traditional computer attacks, ransomware comes to mind. This type of malware has become quite popular among attackers and even has been in the news due to the devastating impact it can have. The popularity of this type of malware and the changes in the automotive industry raise the question of whether there can be a similar attack on vehicles.

The main incentive for creating an automotive ransomware is the possible profitability. Contrary to classical malware on computers, an automotive attack will require substantially more resources to create. To compensate this effort, an automotive ransomware attack has to scale well. This can be done by increasing the number of infections and the amount to pay for each. Here lies the potential of the automotive industry as a ransomware target.

Statistics of the International Organization of Motor Vehicle Manufacturers show that the automotive industry is growing constantly[13] and the average vehicle is continuously shifting towards a modern vehicle[14]. This means that the number of potential targets is increasing. Considering the average cost of a car, the demanded ransom can be significantly higher than the average amount of a traditional ransomware attack.

2 RELATED WORK

The field of automotive security has been researched for several years now. Most of this research focuses on ways to manipulate vehicles and automotive attack vectors in general. In 2010 a team of researchers analyzed and demonstrated how to control various functionality of a vehicle through an attack[5]. These findings give a basis for different manipulations to be potentially utilized by an automotive ransomware attack. Miller and Valasek did an extensive evaluation on vehicle attack surfaces[7] showing many different possible entrypoints which could also be utilized. In 2015, Miller and Valasek demonstrated a remote attack of an unaltered passenger vehicle[11]. The same attack could have been modified into a large-scale state-of-the-art ransomware attack.

The idea of an automotive ransomware has been discussed by ESCRYPT in 2018[6]. Marko Wolf et al. introduced an attack scheme with focus on a Command and Control (C&C) server based ransomware, discussed the possibilities of distribution and created an extensive security concept for building ransomware-proof vehicles. This paper takes a more general approach on automotive ransomware, introduces a model for risk estimation and focuses on

*Produces the permission block, and copyright information

Unpublished working draft. Not for distribution

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM-CSCS'19, September 2019, Munich, Germany
© 2019 Association for Computing Machinery.
ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

an in-depth threat analysis. Furthermore, this paper discusses the proof of concept implementation on a real vehicle.

3 RANSOMWARE

Taking the different types of ransomware explained in Symantec's tech report[15] and also considering doxware[17], one of them seems the most reasonable for automotive attacks. Both crypto ransomware and doxware rely on important or sensitive data, which a usual vehicle does not contain a lot of. Systems like hands-free calling or the camera for driver drowsiness detection could be used to collect sensitive information. Nonetheless, the worst for a car owner to happen is the car being unusable or being a danger to his health. Therefore, this paper focuses on automotive locker ransomware. All further descriptions about ransomware components refer to this type.

3.1 Ransomware components

This section gives an overview on the components for an ideal automotive ransomware. Basis for this list is the systematic segmentation of the traditional ransomware called "WannaCry"[2][9][4] into components. The identified ransomware components have then been put into the automotive context and extended by many automotive-specific components.

Each ransomware component can be influenced by multiple rating metrics. Rating metrics are introduced in section 4. The dependencies between ransomware components and rating metrics allow the creation of a risk model to evaluate the likelihood of an automotive ransomware attack and possible prevention strategies. Every component has an *Importance* property. The importance indicates the necessity of this component for a ransomware implementation. It was categorized by how effective the ransomware will still be if it is not implemented. If a component's importance is very high, it is unlikely to generate any profit through a ransomware attack, when this component isn't implemented. In contrast, for a component rated low, it is imaginable to create a fully functional implementation of a ransomware. Still, the implementation of a component with low importance will increase the possible profit of a ransomware attack.

Initial Infection

Definition Initial entry-point into the car.

Description It grants access to the car's inner network and permanently takes over an internal system.

Importance Very High

Influenced by
- Attack Surface Attackability
- Topology
- Protocol Vulnerability
- Operating System
- Hardware Properties
- Update Mechanism

Self-Distribution

Definition A mechanism to spread the infection onto new targets from an infected target.

Description An automotive ransomware can distribute itself through an outside entity (e.g. repair shop testers[16]) or by using an infected vehicles' communication capabilities and self-replicating across vehicles.

Effect Mass infections are possible.

Importance High

Influenced by
- Attack Surface Attackability
- Topology
- Protocol Vulnerability

Internal Spread

Definition A functionality to infect further internal components of an automotive network.

Description Once inside the car's network, a possibility to spread across many ECUs. This gives access to safety-critical functions and actuators only specific ECUs have. Internal Spread can be a component to escalate privileges inside a virtualized ECU. Additionally, in case an ECU is replaced, it could be infected again.

Effect Increases the impact and difficulty to remove.

Importance Medium

Influenced by
- Topology
- Protocol Vulnerability
- Operating System
- Hardware Properties
- Update Mechanism

Dynamic Attack

Definition An advanced functionality for automated infection modification.

Description Altering an ECUs firmware requires processor architecture specific exploits. They may vary between different firmware versions or different feature configurations of the same car model. A dynamic attack component can update the attack mechanisms of a ransomware based on target identification.

Effect Increases the number of potential targets.

Importance Low

Influenced by
- Operating System
- Hardware Properties
- Number of Vulnerable Vehicles

Unpublished working draft. Not for distribution.

Download Functionality

Definition	Mechanism to download and extend functionality after an infection.
Description	Due to a car's architecture and the heterogeneity of ECU processor architectures, an advanced malware may need a lot of storage for custom attack scripts and exploits. The malware could be stripped down to the bare minimum and download the actual malicious code after the infection through a data connection. This feature goes hand in hand with the Dynamic Attack component.
Effect	Provides extendable functionality with minimum storage requirements.
Importance	Low
Influenced by	• Attack Surface Attackability • Topology • Protocol Vulnerability • Operating System • Update Mechanism

Persistence Mechanism

Definition	A mechanism to lock the ransomware removal possibilities.
Description	In order for the ransomware to be effective it must be hard to remove. All debug, re-flashing, update and restore mechanisms may be disabled. In this case, not even the ECUs Original Equipment Manufacturer (OEM) or repair shops can reset the ECU firmware to remove the ransomware.
Effect	Increases the costs for removal.
Importance	High
Influenced by	• Operating System • Hardware Properties • Update Mechanism

Malicious Activity

Definition	Behaviour forcing the victim to pay the ransom.
Description	Locker ransomware seems very effective. Certain functions or the entire vehicle could be made unusable. In addition, random behaviour and telling the victim about disabled safety features increases the pressure to pay.
Effect	Pressures the user into paying.
Importance	Very High
Example	Preventing the engine from starting.
Influenced by	• Topology • Protocol Vulnerability • Operating System • Vehicle Attractiveness

User Interaction

Definition	A way to communicate with the ransomware's victim.
Description	This component informs the victim about the ransomware infection and gives advice on recovering the car and the payment. Also, the victim is able to perform inputs to interact with the ransomware.
Effect	Notification of the ransomware victim.
Importance	Very High
Example	The infotainment system can be used for interaction. Inputs can be made through existing controls.
Influenced by	• Topology • Protocol Vulnerability • Operating System

Payment

Definition	Payment process, money exchange mechanism.
Description	The payment must not expose the attackers' identities and must be tied to a specific car. Also the amount of the ransom has to be considered. The components User Interaction and Communication may be used for implementing the payment process.
Importance	Very High
Example	This may be achieved by using a cryptocurrency that allows attaching a vehicle specific identifier as a payment message.
Influenced by	• Attack Surface Attackability • Hardware Properties • Attack Surface Scalability • Number of Vulnerable Vehicles • Vehicle Attractiveness

Communication

Definition	A data link for the communication with a C&C server.
Description	This might be used for key exchanges, transmitting IDs, payment confirmations and more. A data connection is needed in order to implement this.
Effect	Allows information exchange with a C&C.
Importance	Low
Influenced by	• Attack Surface Attackability • Topology • Protocol Vulnerability

Unpublished working draft. Not for distribution.

Revert Trigger

Definition A process for initiating the revert mechanism.

Description In order to initiate the revert mechanism, a trigger is needed. It must be implemented in a way that only the attackers approval releases a specific car.

Effect Authenticated ransomware removal.

Importance High

Example On payment the attacker provides a secret for regaining access to the car.

Influenced by
- Attack Surface Attackability
- Topology
- Protocol Vulnerability

Revert Mechanism

Definition A mechanism to uninstall a ransomware.

Description Victims have to be certain that after unlocking everything will be as safe as without the ransomware, otherwise they might not pay. A malfunctioning car as a result of an incomplete removal can be life-threatening. Even after the ransomware removal driving might be against the law and insurance companies might refuse to pay in case of an accident. Additionally the car's warranty might be gone. This may lead to the entire car having to be reprogrammed or certain components need to be replaced.

Effect Removal of a ransomware. Reward for paying.

Importance High

Influenced by
- Operating System
- Hardware Properties
- Update Mechanism

Hardening Mechanism

Definition Mechanisms to increase the difficulty for reverse engineering and countermeasures.

Effect Increase the ransomware resilience.

Importance Medium

Example Disabling ways to read the ransomware's code from the ECU is one example. The ransomware using encrypted messages within the vehicle is another one. Also a mechanism to stop any bad behaviour when a virtual environment or a development board is detected would counter examinations within a simulated environment.

Influenced by
- Operating System
- Hardware Properties
- Update Mechanism

Anti-Reinfection

Definition Prevent infections of a vehicle that was already paid for.

Description Reinfections after the payment may make the victims lose trust of regaining the car. A mechanism to prevent another infection is required.

Effect Increases trust in the ransomware payment.

Importance Medium

Influenced by
- Operating System
- Hardware Properties

4 RATING METRICS FOR RISK EVALUATION OF AUTOMOTIVE RANSOMWARE

These metrics are the core components of the introduced risk evaluation model. The model is the result of an evaluation of difficulties during the implementation of an automotive ransomware. It is based on a general understanding of a car's structure, several research papers[3][10][7] and on our own proof of concept implementation.

It takes basic vehicle properties of a modern vehicle and puts them into context for a risk estimation on how prone the vehicle is to ransomware attacks. These properties are known by car manufacturers or the necessary information can be acquired through inspecting the vehicle of interest. When estimating the risk, the attack's scalability and the effort for the attack creation have to be taken into account. Unless attackers want to directly harm a specific entity, these are the factors they most likely consider.

Attack Surface Attackability

Description Attack Surface in terms of attackability measures the quality of an attack surface to be utilized by a ransomware.

Vehicle Properties
- **Likelihood of unauthorized or insecure devices being connected** considers devices like smartphones or On-board diagnostics (OBD-II) dongles
- Insecure **back end servers** can be used as attack vector
- **Critical features on interfaces** increase the possibilities for exploits
- **Fewer layers of separation from the attack surface to the vehicle inner network** lead to less steps for accessing critical functionality
- Attack vectors over a large **distance** lead to less restrictions of attacks because there is no need to be physically close to the target
- The possibility of **temporary or permanent data connection** allows C&C server communication

Ransomware Components
- Initial Infection
- Self-Distribution
- Download Functionality
- Payment
- Communication
- Revert Trigger

Unpublished working draft. Not for distribution.

Topology

Description Topology measures the ease of attacking a vehicle by considering the vehicle's network layout.

Vehicle Properties

- Through a high **variety in network-technologies** an attack will be significantly harder
- **Isolation of critical ECUs** onto separated network domains makes it necessary to infect several networks for advanced malicious activity
- Presence of a **gateway** or a **firewall** between networks increases the effort to spread attacks between networks and protects the internal network from outside attack surfaces
- ECUs that **bridge networks** may allow for bypassing the network separation
- **Intrusion Detection System (IDS) or Intrusion Prevention System (IPS)** might prevent the infiltration of safety-critical networks

Ransomware Components

- Initial Infection
- Self-Distribution
- Internal Spread
- Download Functionality
- Malicious Activity
- User Interaction
- Communication
- Revert Trigger

Protocol Vulnerability

Description Protocol Vulnerability measures the ease of manipulating the communication.

Vehicle Properties

- If the **used protocols** are **common with other vehicles** more generic attacks may be possible
- **Criticality of communication** considers messages to be misused for malicious activity
- Both **authenticated** and **encrypted communication** increase the difficulty to manipulate the communication

Ransomware Components

- Initial Infection
- Self-Distribution
- Internal Spread
- Download Functionality
- Malicious Activity
- User Interaction
- Communication
- Revert Trigger

Operating System

Description The Operating System metric measures the ease of attacking an ECU considering operating system related vehicle properties.

Vehicle Properties

- A **well known** Operating System (OS) (e. g. a unix-like system) can be easier to attack because attackers may already be familiar with it and its vulnerabilities
- If an OS or a software component is **widely used** in the automotive industry, more ECUs may be affected by the same vulnerability
- If the OS offers **high abstraction**, easier and more generic attacks may be possible
- If an OS is implemented and **designed securely**, it might contain less exploitable security flaws
- **Virtualization or separation** reduces the possibilities for privilege escalation

Ransomware Components

- Initial Infection
- Internal Spread
- Dynamic Attack
- Download Functionality
- Persistence Mechanism
- Malicious Activity
- User Interaction
- Revert Mechanism
- Hardening Mechanism
- Anti-Reinfection

Hardware Properties

Description Hardware Properties measures the attackability by taking the ECUs hardware capabilities and properties into account.

Vehicle Properties

- With an ECU having high **processing power** some computation-intensive attack scenarios may become realizable
- Only if **persistent memory access** is available, a ransomware is able to permanently infect an ECU
- A large **variety in hardware architectures** results in the demand of individual exploits and compilations for each architecture
- With a larger **number of ECUs**, more individual targets may have to be attacked
- If there are several **different firmware versions** for the same ECU in use, an attack might have to be customized depending on the firmware version.

Ransomware Components

- Initial Infection
- Internal Spread
- Dynamic Attack
- Persistence Mechanism
- Payment
- Revert Mechanism
- Hardening Mechanism
- Anti-Reinfection

Unpublished working draft. Not for distribution.

Update Mechanism

Description Update Mechanism measures the likelihood of the update ecosystem to be misused for altering the firmware.

Vehicle Properties

- **Over-The-Air (OTA)** updates change the approach for misusing the update mechanism compared to updates via bootloader.
- The exploit will be easier if a **common bootloader** is used. Flashing mechanism might be standardized and therefore the exploit might be applicable on several different ECUs.
- Having the **firmware available online** for download means that it is easily acquirable by attackers. This simplifies the reverse engineering process since attackers do not need to extract firmwares from ECUs.
- This is not the case when the firmware image is **encrypted**
- An update mechanism only accepting **cryptographically signed** firmwares increases the difficulty of manipulating the firmware.

Ransomware Components

- Initial Infection
- Internal Spread
- Download Functionality
- Persistence Mechanism
- Revert Mechanism
- Hardening Mechanism

Attack Surface Scalability

Description Attack Surface in terms of scalability measures the quality of attack surfaces for large-scale attacks.

Vehicle Properties

- Long-range attacks over a large **distance** are able to target more vehicles.
- The **reachability and availability** of a potential attack vector is considered because of features disabled by default or ones that only work with the vehicle turned on.
- If the **infection is fully automatable** without the need of an attacker's or car owner's manual operation, a large-scale attack will be easier to execute.

Ransomware Components

- Payment

Number of Vulnerable Vehicles

Description Number of Vulnerable Vehicles measures the number of possible targets.

Vehicle Properties

- **Commonness of architectures used** considers architecture-based vulnerabilities that affect not only a single car model.
- Similarly, if the **ECUs are widespread** and reused in many different vehicles, those may also be prone to an attack.
- **Over-The-Air updates** decrease the number of vulnerable vehicles if the vulnerabilities are fixed.

Ransomware Components

- Dynamic Attack
- Payment

Vehicle Attractiveness

Description Vehicle Attractiveness measures the impact on the payout through a car's attributes.

Vehicle Properties

- A large **number of ECUs** allows for a more complex infection which increases the cost for removal at a repair shop
- If the average **repair cost** is high, the ransom to be paid may be higher
- A higher **car value** also increases the ransom that can be demanded
- A large **number of advanced features** to be misused for malicious behavior increases the possibilities to threaten the car owner
- If a vehicle contains **functionality possible to be misused to collect sensitive data**, a doxware-type automotive ransomware could be created
- **Importance of availability** considers vehicles that are used frequently and can not be replaced easily

Ransomware Components

- Malicious Activity
- Payment

Unpublished working draft.
Not for distribution.

4.1 Risk

The general likelihood of a specific vehicle being targeted with an automotive ransomware attack can be estimated through the introduced model. The combination of the introduced components with vehicle specific rating metrics provide a structured approach for this. Each rating metric can be evaluated for a vehicle by evaluating each influencing property with a checklist procedure. The combination of all rating metrics associated with a ransomware component gives an estimation on the difficulty of its implementation. All components with a high or very high importance need to be implemented for a functioning ransomware. If the average difficulty for the implementation of these necessary components is low, a ransomware attack is more likely. Components of low importance increase the impact of a ransomware attack. These components will increase the possible profit of an attacker.

5 PROOF OF CONCEPT IMPLEMENTATION

To prove this model, a basic ransomware was implemented for a real car. The implementation of this ransomware created insights in the implementation difficulties of a real world automotive ransomware. In the context of responsible disclosure we choose to not provide details on the exploit or the specific car used as it may endanger car owners. First, we state all rating metrics derived from the vehicle properties. This evaluation gives us knowledge about the implementation difficulty of a specific component. Lastly, we explain the implementation of all individual components and give a rating for the difficulty of the implementation. The estimated implementation difficulty will be compared with real implementation difficulty to prove this model.

5.1 Vehicle specific rating metrics

The following metrics were derived based on the properties of a real car.

Attack Surface Attackability **High**

The investigation object has several attack surfaces (LTE/GSM, WiFi, Bluetooth, OBD-II). Critical features, for example unprotected flashing mechanisms, are available without authentication on OBD-II. Internal ECUs are not protected through a network architecture with separated domains. Large distance attack vectors are not vulnerable for well known exploits. Attacks on low-distance interfaces (e. g. OBD-II) are easy to implement. The car's cellular communication can be utilized for a permanent data connection.

Topology **Very High**

No central gateway. All internal networks are connected to the OBD-II connector. Safety-critical ECUs share a communication bus with remotely attackable ECUs. Some vulnerable ECUs can be used as a bridge between networks. This network topology makes the vehicle very prone to attacks.

Protocol Vulnerability **High**

The same protocols are used for different car brands. Safety-critical commands are available in the protocol and can be abused by an attacker. Weak authentication and no encryption are present. An attacker can fake or replay any Controller Area Network (CAN) message.

Operating System **Low**

Most ECUs are using a proprietary real time OS. Only two ECUs are using the unix-like OS QNX. The absence of a common function set for every ECU leads to customized attacks for every ECU which increases the effort for an attacker. Virtualized ECUs are not present in this car.

Hardware Properties **Low**

Most ECUs have a comparable low processing power. Every ECU is designed for a single purpose. Hardware-Watchdogs will reset ECUs if the Central Processing Unit (CPU) utilization is too high. A large variety of processor architectures increases the difficulty of an attack. Since hardware-watchdogs supervise the CPU utilization of safety-critical ECUs, exploits have to be crafted very carefully.

Update Mechanism **High**

Over-The-Air update mechanism are present but not enabled. ECU updates are not cryptographically signed. Firmware images are not encrypted. Update mechanism can be exploited over the vehicle internal networks.

Attack Surface Scalability **Low**

Large-Distance attack surfaces are present. A built-in Telematic Communication Unit (TCU) provides a permanent data connection to a backend system. Exploitation of large-distance attack vectors is very difficult.

Number of Vulnerable Vehicles **High**

Value priced car. ECUs are identical on many different brands of the same OEM. Only small changes are required to port the implementation of a ransomware to different car models or brands from the same OEM.

Vehicle Attractiveness **Very High**

Absence of cryptographically signed firmware images gives an attacker all possibilities on malicious activities. Any actuator of the vehicle could potentially be used to frighten the owner of the car.

5.2 Implementation of Ransomware components

This section describes, how the components were implemented on a real car. A rating for the effort of the implementation of a specific component is given as well. If a component could be implemented in short time or without the necessity to overcome any protection mechanism the effort was rated very low to low whereas if the opposite was the case, we rated it high to very high.

Initial Infection **Medium**

The software update mechanism over OBD-II was used for the initial infection. A vulnerable repair shop tester [16] or a vulnerable OBD-II dongle [8] can distribute an initial ransomware infection to the car.

Self-Distribution **-**

This component was not implemented for the proof of concept.

Internal Spread **High**

Since the software update mechanism of this car was vulnerable, the internal spread component could have been implemented through the same software update mechanism, already used for the initial infection. To achieve an internal spread, exploits for further ECUs would need to be included in the ransomware. This would increase the size and the complexity of this ransomware.

Dynamic Attack **-**

This component was not implemented since the ransomware was planted in only one ECU.

Download Functionality **Very High**

The targeted ECU did not have any communication capabilities with a remote server. An implementation of this would become possible by forwarding the data connection of the communication component.

Persistence Mechanism **Medium**

Since the initial infection modified the firmware image of the targeted ECU, this component could have been implemented easily. A modification of the security access mechanisms would be sufficient to lock out the OEMs repair shop tools. This would stop a repair shop from removing the ransomware through reprogramming the infected ECU. At this point, a hardware replacement and the man-hour costs of a repair shop are required to remove the ransomware from a car.

Malicious Activity **Very Low**

Repair shop tester functions were used to implement this component. These functions are used to trigger specific functions during a car repair. Our ransomware was able to abuse these commands by sending requests to various ECUs over the inner network.

User Interaction **Low**

The proof of concept ransomware was able to abuse a service used for displaying WiFi settings to the car owner. This service allows the ransomware to display text messages on the multimedia screen of the car. Button presses on the touch screen were sent on the vehicle internal CAN bus. The ransomware, running on a different ECU, achieves a bi-directional communication just by sending and receiving CAN and CAN-Layer Transport Protocol (ISOTP) messages.

Payment **Low**

To process the payment, the user was provided with a bitcoin account number through the multimedia interface. A way to tie the payment to a vehicle was not implemented for the proof of concept ransomware.

Communication **Very High**

Data communication to a backend server was not possible to be implemented during the proof of concept development. This component would require an attack of the TCU of this car.

Revert Trigger **-**

A revert trigger was not necessary for the proof of concept implementation.

Revert Mechanism **Low**

The ransomware execution was achieved through a hijack of interrupt vectors in the ECU firmware. A simple restore of the original interrupt vectors in the program memory of the ECU was sufficient to revert the ransomware.

Hardening Mechanism **-**

Hardening mechanisms were not implemented on the proof of concept ransomware.

Anti-Reinfection **-**

This component was not necessary for the proof of concept. The simplest implementation would be to fix the vulnerability.

Unpublished working draft. Not for distribution.

5.3 Comparison between model and implementation

The effort for the implementation of the most important components of a ransomware is compared to the difficulty derived from the model. This shows how our model can be applied to identify critical vehicle properties. An estimated effort for the implementation of a component is indicated by the combination of all vehicle metrics influencing this component.

Initial Infection

The estimated effort for an initial infection is medium. Also the effort for a real implementation is medium. The difficulty for this implementation is lowered through the metrics Attack Surface Attackability, Protocol Vulnerability, Update Mechanism and Topology. Therefore, a secured update mechanism and signed firmware updates would increase the difficulty of an implementation. An initial infection would not be possible with medium effort if these countermeasures would be present in the car.

Persistence Mechanism

Considering the model, the effort for the implementation of the Persistence Mechanism is medium to high. We rate the difficulty for the implementation medium. The weak security of the update mechanism made it possible. Again, a very effective countermeasure to prevent the implementation of this component would be the use of signed firmwares.

Malicious Activity

Evaluating the model indicates very low effort to implement the component Malicious Activity. During the implementation of the real world ransomware, very low effort was required to implement this component. Changes in the vehicle internal network topology and authenticated communication for repair shop testers would increase the difficulty for the implementation of this component.

User Interaction

Through the combination of all metrics influencing the component User Interaction, very low effort for the implementation was predicted. For the implementation in a real car, low effort was spent. Authenticated internal communication would be a sufficient mitigation to raise the difficulty for this component. Since this component is crucial for a ransomware, the likelihood of a ransomware attack could be lowered through this countermeasure extensively.

Payment

The estimated difficulty for the implementation of the Payment component is low to very low. Our proof of concept implementation was not fully functional but could have been extended with little effort to tie the payment to the car. Since an attacker could use the bitcoin cryptocurrency for the payment process, a real world implementation would require low effort. In order for this component to have effect, it requires the User Interaction component. If an attacker can not show instructions for the payment, the implementation of this component is not possible. Therefore this is the easiest way to increase the difficulty for a payment process.

Revert Mechanism

Our model predicted a medium difficulty for the implementation of the component Revert Mechanism. Through the absence of signed firmware images for ECUs, the effort spent on the implementation of this component was low.

5.4 Summary

Using encrypted communication and a signed update mechanism would have increased the difficulty for implementing many crucial ransomware components. Therefore, with very few changes, the risk could be decreased dramatically. Analyzing the metrics influencing the components with the highest importance allows for finding the weaknesses with the highest impact in the vehicles' design. This allows for well-aimed countermeasures.

6 CONCLUSION

Our proof of concept implementation shows that, from a technical point of view, automotive ransomware attacks are possible in real world scenarios. It is likely that an automotive ransomware attack has the potential to scale very well. A locker ransomware is the most expectable type of ransomware for automotive systems. In addition to ransomware components that are absolutely necessary there are optional ones that increase the impact. Basic factors that influence the risk of a specific vehicle were found and put together as vehicle specific rating metrics. As shown, the introduced model can be used to identify problematic vehicle properties. Through the correlation between vehicle properties and ransomware components, a car manufacturer can identify effective countermeasures against specific components. These insights can be used for creating a security concept and security architecture within the security extended V-model[18][1]. This way, the vehicle's design can be improved in terms of the ransomware threat and overall.

7 FUTURE WORK

With time passing and new technologies being developed it may be necessary to extend or adjust the automotive ransomware components introduced in section 3.1. This also means that the model may have to be extended by new rating metrics. Extensive research on many different vehicles and a deep evaluation of all vehicle properties may allow a unified rating of vehicles. This abstract model is a first step for a unified risk estimation model which allows comparisons of the security level of different vehicles.

Additional contributions to this work includes the research and demonstration of a mass infection of many ECUs. An analysis on how different attack surfaces can be utilized for automotive ransomware may be worth investigating. Even though there are already researches focusing on attack surfaces [7][11][10], they do not consider any special requirements an automotive ransomware might have.

Lastly a security concept or a maturity model can be created based on this paper's findings.

REFERENCES

[1] Cornelius Bittersohl and Timothy G. Thoppil. [n. d.]. *Automotive Cybersecurity. Developing a thriving security ecosystem within automotive.* P3 North America, Inc.

Unpublished working draft. Not for distribution.

[2] CERT-MU. 2017. *THE WANNACRY RANSOMWARE.* Technical Report. CERT-MU.
[3] Computest. 2018. The Connected Car: Ways to get unauthorized access and potential implications. (2018).
[4] ENDGAME. 2017. WCry/WanaCry Ransomware Technical Analysis. (2017). Retrieved January 10, 2019 from https://www.endgame.com/blog/technical-blog/wcrywanacry-ransomware-technical-analysis
[5] Karl Koscher et al. 2010. Experimental Security Analysis of a Modern Automobile. (2010).
[6] Marko Wolf et al. 2018. WANNA DRIVE? Feasible Attack Paths and Effective Protection Against Ransomware in Modern Vehicles. (2018).
[7] Stephen Checkoway et al. 2011. Comprehensive Experimental Analyses of Automotive Attack Surfaces. (2011).
[8] Rudolf Hackenberg, Nils Weiss, Sebastian Renner, and Enrico Pozzobon (Eds.). 2017. *Extending Vehicle Attack Surface Through Smart Devices.* IARIA.
[9] LogRhythm Labs. 2017. A Technical Analysis of WannaCry Ransomware. (2017). Retrieved January 7, 2019 from https://logrhythm.com/blog/a-technical-analysis-of-wannacry-ransomware/
[10] Dr. Charlie Miller and Chris Valasek. 2014. A Survey of Remote Automotive Attack Surfaces. (2014).
[11] Dr. Charlie Miller and Chris Valasek. 2015. Remote Exploitation of an Unaltered Passenger Vehicle. (2015).
[12] Dr. Charlie Miller and Chris Valasek. 2016. CAN Message Injection. (2016).
[13] International Organization of Motor Vehicle Manufacturers. 2015. Motorization rate 2015 - WORLDWIDE. (2015). Retrieved January 11, 2019 from http://www.oica.net/category/vehicles-in-use/
[14] International Organization of Motor Vehicle Manufacturers. 2017. 2005-2017 Sales Statistics. (2017). Retrieved January 11, 2019 from http://www.oica.net/category/sales-statistics/
[15] Kevin Savage, Peter Coogan, and Hon Lau. 2015. *The evolution of ransomware.* Technical Report. Symantec. Version 1.0.
[16] András Szijj, Levente Buttyán, and Zsolt Szalay. 2015. Hacking cars in the style of Stuxnet. (2015). Retrieved May 21, 2019 from http://www.hit.bme.hu/~buttyan/publications/carhacking-Hacktivity-2015.pdf
[17] Brian Thorne. 2017. Computer Security: Enter the next level: Doxware. (2017). Retrieved May 14, 2019 from http://cds.cern.ch/record/2291225
[18] Dr. Markus Tschersich. 2018. *Cybersecurity in the Automotive Domain. PWIN Guest Lecture.* Continental AG.

Unpublished working draft
Not for distribution